Funktionsbeschreibung Ausgabe 07/2004

sinumerik

SINUMERIK 840D/840Di/810D
ISO-Dialekte für SINUMERIK

SIEMENS

SIEMENS

SINUMERIK 840D/840Di/810D

ISO–Dialekte für SINUMERIK

Funktionsbeschreibung

Gültig für

Steuerung	*Softwarestand*
SINUMERIK 840D powerline	7
SINUMERIK 840DE powerline	7
SINUMERIK 840Di	3
SINUMERIK 840DiE (Exportvariante)	3
SINUMERIK 810D powerline	7
SINUMERIK 810DE powerline	7

Ausgabe 07.2004

SINUMERIK®–Dokumentation

Auflagenschlüssel

Die nachfolgend aufgeführten Ausgaben sind bis zur vorliegenden Ausgabe erschienen.

In der Spalte ”Bemerkung” ist durch Buchstaben gekennzeichnet, welchen Status die bisher erschienen Ausgaben besitzen.

Kennzeichnung des Status in der Spalte ”Bemerkung”:

A . . . . . Neue Dokumentation.
B . . . . . Unveränderter Nachdruck mit neuer Bestell–Nummer
C . . . . . Überarbeitete Version mit neuem Ausgabestand.
Hat sich der auf der Seite dargestellte technische Sachverhalt gegenüber dem vorherigen Ausgabestand geändert, wird dies durch den veränderten Ausgabestand in der Kopfzeile der jeweiligen Seite angezeigt.

Ausgabe	Bestell–Nr.	Bemerkung
08.99	6FC5297–5AE10–0AP0	**A**
04.00	6FC5297–5AE10–0AP1	**C**
10.00	6FC5297–6AE10–0AP0	**C**
09.01	6FC5297–6AE10–0AP1	**C**
12.01	6FC5297–6AE10–0AP2	**C**
11.02	6FC5297–6AE10–0AP3	**C**
07.04	6FC5297–6AE10–0AP4	**C**

Dieses Buch ist Bestandteil der Dokumentation auf CD–ROM **(DOCONCD)**

Ausgabe	Bestell–Nr.	Bemerkung
09.04	6FC5 298–7CA00–0AG1	**C**

Marken
SIMATIC®, SIMATIC HMI®, SIMATIC NET®, SIROTEC®, SINUMERIK® und SIMODRIVE® sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Druckschrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen können.

Weitere Informationen finden Sie im Internet unter:
http://www.siemens.de/motioncontrol

Die Erstellung dieser Unterlage erfolgte mit Interleaf V 7

Es können weitere, in dieser Dokumentation nicht beschriebene Funktionen in der Steuerung lauffähig sein. Es besteht jedoch kein Anspruch auf diese Funktionen bei Neulieferung bzw. im Servicefall.

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard– und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden. Die Angaben in dieser Druckschrift werden jedoch regelmäßig überprüft, und notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten. Für Verbesserungsvorschläge sind wir dankbar.

Technische Änderungen vorbehalten.

Bestell–Nr. 6FC5 297–6AE10–0AP4
Printed in the Federal Republic of Germany

Siemens–Aktiengesellschaft

Vorwort

Gliederung der Dokumentation

Die SINUMERIK–Dokumentation ist in 3 Ebenen gegliedert:

- Allgemeine–Dokumentation
- Anwender–Dokumentation
- Hersteller/Service–Dokumentation

Nähere Informationen zu Druckschriften über SINUMERIK 840D/840Di/810D sowie Druckschriften, die für alle SINUMERIK–Steuerungen gelten, erhalten Sie von Ihrer SIEMENS–Niederlassung.

Adressat

Die vorliegende Dokumentation wendet sich an den Hersteller von Werkzeugmaschinen mit SINUMERIK 840D bzw. SINUMERIK 810D und SIMODRIVE 611D.

Hotline

Bei Fragen wenden Sie sich bitte an folgende Hotline:

A&D Technical Support
Tel.: +49 (0)180 5050 222
Fax: +49 (0)180 5050–223
email: adsupport@ad.siemens.com

Bei Fragen zur Dokumentation (Anregungen, Korrekturen) senden Sie bitte ein Fax an folgende Faxadresse:
Fax: +49 (0) 9131 / 98–2176
E–Mail: motioncontrol.docu@erlf.siemens.de

Faxformular: siehe Rückmeldeblatt am Schluß der Druckschrift

Internetadresse

http://www.siemens.de/motioncontrol

SINUMERIK 840D powerline

Seit 09.2001 steht die

- SINUMERIK 840D powerline und
- SINUMERIK 840DE powerline

mit verbesserter Performance zur Verfügung. Eine Auflistung der verfügbaren **powerline**–Baugruppen finden Sie in der Hardware–Beschreibung:

Literatur: /PHD/, Handbuch Projektierung SINUMERIK 840D

Vorwort

SINUMERIK 810D powerline

Seit 12.2001 steht die

- SINUMERIK 810D powerline und
- SINUMERIK 810DE powerline

mit verbesserter Performance zur Verfügung. Eine Auflistung der verfügbaren **powerline**–Baugruppen finden Sie in der Hardware–Beschreibung:

Literatur: /PHC/, Handbuch Projektierung SINUMERIK 810D

Addressat des Handbuchs

- Projekteure
- Elektriker und Monteure
- Service– und Betriebspersonal

Gegenstand des Handbuches

Die Informationen dieses Handbuches ermöglichen es, Teileprogramme von externen CNC–Systemen einzulesen und abzuarbeiten.

Suchhilfen

Zu Ihrer besseren Orientierung wird Ihnen ein Inhaltsverzeichnis angeboten sowie die Anhänge:

- Literaturverzeichnis
- Stichwortverzeichnis
- Befehlsverzeichnis

Vorwort

Gefahr- und Warnkonzept

In dieser Druckschrift werden folgende Gefahr- und Warnhinweise verwendet. Hier die Symbolerläuterung:



Gefahr

Dieser Warnhinweis bedeudet, daß Tod schwere Körperverletzung oder erheblicher Sachschaden eintreten **werden**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Warnung

Dieser Warnhinweis bedeutet, daß Tod, schwere Körperverletzung oder erheblicher Sachschaden eintreten **können**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.



Vorsicht

Dieser Warnhinweis (mit Warndreieck) bedeutet, daß eine leichte Körperverletzung oder ein Sachschaden eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Vorsicht

Dieser Warnhinweis (ohne Warndreieck) bedeutet, daß ein Sachschaden eintreten **kann**, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

Achtung

Dieser Warnhinweis bedeutet, daß ein unerwünschtes Ereignis oder ein unerwünschtes Zustand eintreten **kann**, wenn der entsprechende Hinweis nicht beachtet wird.

Weitere Hinweise



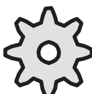
Wichtig

Dieser Hinweis bedeutet, daß ein wichtiger Sachverhalt zu beachten ist.

Hinweis

Dieses Symbol erscheint in dieser Dokumentation immer dann, wenn weiterführende Sachverhalte angegeben werden.

Vorwort



Maschinenhersteller

Das abgebildete Symbol erscheint in dieser Dokumentation immer dann, wenn der Maschinenhersteller das beschriebene Funktionsverhalten beeinflussen oder verändern kann. Beachten Sie die Angaben des Maschinenherstellers.

Warenzeichen

IBM ist ein eingetragenes Warenzeichen der International Business Corporation. MS DOS und WINDOWS™ sind eingetragene Warenzeichen der Microsoft Corporation.

Inhalt

Platz für Notizen

Kurzbeschreibung 1

Einführung

Es können Teileprogramme von externen CNC-Systemen eingelesen und abgearbeitet werden.

Diese Druckschrift beschreibt die dafür notwendige Inbetriebnahme und das Abarbeiten von NC-Programmen eines externen CNC-System. Darüber hinaus werden auch funktionale Unterschiede erläutert.

Hinweis

Eine ausführliche Beschreibung der externen Programmierfunktionen finden Sie in der Original-Dokumentation des externen CNC-Systems.

Verwendete Begriffe

Für diese Druckschrift wurden folgende Begriffe festgelegt:

- ISO–Dialekt–M ist ähnlich dem G–Code der Steuerung ”Fanuc16–Milling”
- ISO–Dialekt–T ist ähnlich dem G–Code der Steuerung ”Fanuc16–Turning” System B
- ISO–Dialekt–Original entspricht der Original–Fanuc16-Steuerung

■

Platz für Notizen

Programmierung

2

2.1 Aktivierung der Funktionen

Mit dem Maschinendatum 18800 $MN_EXTERN_LANGUAGE wird die externe Sprache aktiviert. Die Auswahl des Sprachtyps ISO-Dialekt-M oder T erfolgt über das Maschinendatum 10880 $MN_EXTERN_CNC_SYSTEM.
Die externe Sprache kann getrennt für jeden Kanal aktiviert werden. Zum Beispiel kann Kanal 1 im ISO-Mode laufen, während Kanal 2 im Siemens-Mode aktiv ist.

Umschaltung

Die Umschaltung von Siemens-Mode nach ISO-Dialekt-Mode erfolgt durch die beiden G-Befehle aus der Gruppe 47:

- G290 NC-Programmiersprache Siemens aktiv
- G291 NC-Programmiersprache ISO-Dialekt aktiv

Dabei bleiben das aktive Werkzeug, die Werkzeugkorrekturen und Nullpunktverschiebungen erhalten (siehe Kapitel 2.2.4 und 2.4).

Siemens-Mode

Ist der Siemens-Mode aktiv, gelten folgende Bedingungen:

- Standardmäßig werden in der Steuerung G-Befehle von Siemens interpretiert.
- Eine Erweiterung der Siemens-Programmierung um ISO-Dialekt-Funktionen ist nicht möglich, weil ein Teil der G-Funktionen unterschiedliche Bedeutung hat.
- Mit Hilfe von nachladbaren MD-Files kann die Steuerung in den ISO-Dialekt-Mode gesetzt werden. Dadurch sieht der Anwender standardmäßig den ISO-Dialekt-Mode.

ISO-Dialekt-Mode

Ist der ISO-Dialekt-Mode aktiv, gelten folgende Bedingungen:

- Es können nur G-Codes im ISO-Dialekt, keine Siemens G-Codes programmiert werden
- Eine Mischung aus ISO-Dialekt-Code und Siemens-Code in einem NC-Satz ist nicht möglich
- Eine Umschaltung über G-Befehle zwischen ISO-Dialekt-M und ISO-Dialekt-T ist nicht möglich
- Sollen weitere Siemens-Funktionen genutzt werden, muß zuerst in den Siemens-Mode umgeschaltet werden (Ausnahme: Programmverzweigungen und Unterprogrammaufrufe, siehe Kapitel 2.3.2)

PowerOn/Reset

Die Tabelle 10-1 zeigt die möglichen Kombinationen der MD 10880: $MN_MM_EXTERN_CNC_SYSTEM und 20150: $MC_GCODE_RESET_VALUE. Damit wird das Power On/Reset Verhalten festgelegt.

Tabelle 2-1 Aktivierung der Funktionen

Nach Power On/Reset...	$MC_GCODE_RESET_ VALUES[46] =	$MN_EXTERN_CNC_ SYSTEM =
Siemens-Mode aktiv, Umschaltung in ISO-Dialekt-M möglich	**1** G290 Siemens-Mode	**1** ISO-Dialekt-M
Siemens-Mode aktiv, Umschaltung in ISO-Dialekt-T möglich	**1** G290 Siemens-Mode	**2** ISO-Dialekt-T
ISO-Dialekt-M aktiv, Umschaltung in Siemens-Mode möglich	**2** G291 ISO-Dialekt-Mode	**1** ISO-Dialekt-M
ISO-Dialekt-T aktiv, Umschaltung in Siemens-Mode möglich	**2** G291 ISO-Dialekt-Mode	**2** ISO-Dialekt-T

Modal wirksame G-Befehle

Modal wirksame G-Befehle, die in beiden Systemen (Siemens und ISO-Dialekt) funktional identisch sind, werden folgendermaßen behandelt:
Bei der Programmierung dieser G-Codes in der einen Sprache wird der korrespondierende G-Code der anderen Sprache ermittelt und aktiviert. Folgende G-Codes sind davon betroffen:

Datenhaltung

ISO-Programme können im Lochstreifenformat in den MMC103 eingelesen und ausgegeben werden.
Eingelesene ISO-Programme werden in der NC-Datenhaltung als Hauptprogramm abgelegt, dabei defaultmäßig in dem Pfad:
_N_WKS_DIR/_N_SHOPMILL_WPD.

Ein Ändern des Eintrages ist durch Editieren der Datei DINO.INI im Verzeichnis USER möglich. Weitere Informationen sind der Druckschrift

Literatur: /IAM/, IM3: Inbetriebnahmeanleitung MMC. Kapitel 3.1 zu entnehmen.

2.1.1 Umschalten von ISO–Mode nach Siemensmode

G290/291

Über die G–Befehle 290/291 kann aus dem Teileprogramm heraus der Mode gewechselt werden.
Mit der Umschaltung erfolgt auch ein Wechsel der aktuellen G–Code Anzeige

G65/66

Makro satzweise und modal:
Es wird das programmierte Unterprogramm aufgerufen. Eine Umschaltung in den Siemens Mode erfolgt nur, wenn im Unterprogramm in der ersten Zeile die PROC Anweisung verwendet wird.
Wird ein solches Programm mit M17 oder RET beendet, erfolgt beim Rücksprung eine Modeumschaltung zurück in den ISO–Mode.

Siemens Up-Aufruf im ISO-Mode

Modale und nicht modale Unterprogrammaufrufe, z.B.

```
    N100 CALL “WELLE”
oder
    N100 MCALL WELLE
oder
    N100 WELLE
```

Modale und nicht modale Unterprogrammaufrufe mit Parameterübergabe

```
   N100 MCALL WELLE(”ABC”, 33.5)
oder
   N100 WELLE(”ABC”, 33.5)
```

Unterprogrammaufrufe mit Pfadangabe

```
   N100 CALL “/_N_SPF_DIR/WELLE
oder
   N100 MCALL /_N_SPF_DIR/WELLE
oder
   N100 PCALL /_N_SPF_DIR/WELLE
```

Bei Unterprogrammaufrufen wird implizit der Siemens-Mode angewählt und mit dem Unterprogrammende wieder in den ISO-Dialekt-Mode zurück gewechselt.

Modale, satzweise Zyklen

Wird im ISO–Mode ein modaler oder satzweiser Zyklus programmiert, erfolgt der Aufruf eines Hüllzyklus.
Bei diesem Aufruf erfolgt eine Mode–Umschaltung in den Siemensmode.

2.2 G-Befehle

Die G-Codes von ISO-Dialekt-T beziehen sich auf das G-Code System B (siehe auch 4.1.5).

Die aktiven G-Codes des ISO-Modes können über die Systemvariable $P_EXTGG[...] gelesen werden. Die Zahlen neben dem G-Code geben den jeweiligen Wert in $P_EXTGG[...] an. Über das Maschinendatum 20154 EXTERN_GCODE_RESET_VALUES[n]: 0, ..., 30 werden die G-Codes festgelegt, die im Hochlauf wirksam werden, wenn der NC-Kanal nicht im Siemens-Mode läuft.

Tabelle 2-2 Standardeinstellung ist mit 1) gekennzeichnet

ISO-Dialekt-T		ISO-Dialekt-M		Beschreibung
Gruppe 1				
G00 1)	1	G00 1)	1	Eilgang
G01	2	G01	2	Linearbewegung
G02	3	G02	3	Kreis/Helix im Uhrzeigersinn
		G02.2	6	Evolvente im Uhrzeigersinn
G03	4	G03	4	Kreis/Helix gegen den Uhrzeigersinn
		G03.2	7	Evolvente im Gegenuhrzeigersinn
G33	5	G33	5	Gewindeschneiden mit konstanter Steigung
G34	9			Gewindeschneiden mit variabler Steigung
G77	6			Längsdrehzyklus
G78	7			Gewindeschneidzyklus
G79	8			Planflächendrehzyklus
Gruppe 2				
		G17 1)	1	XY-Ebene
		G18	2	ZX-Ebene
		G19	3	YZ-Ebene
G96	1			Konstante Schnittgeschwindigkeit ein
G97 1)	2			Konstante Schnittgeschwindigkeit aus
Gruppe 3				
G90 1)	1	G90 1)	1	Absolute Programmierung
G91	2	G91	2	Inkrementelle Programmierung
Gruppe 4				
		G22	1	Arbeitsfeldbegrenzung, Schutzbereich 3 ein
		G23 1)	2	Arbeitsfeldbegrenzung, Schutzbereich 3 aus
G68	1			Doppelrevolver/-schlitten ein
G69 1)	2			Doppelrevolver/-schlitten aus

Tabelle 2-2 Standardeinstellung ist mit 1) gekennzeichnet

ISO-Dialekt-T	ISO-Dialekt-M	Beschreibung
Gruppe 5		
	G93 3	zeitreziproker Vorschub (1/min)
G94 1	G94 1) 1	Vorschub in [mm/min, inch/min]
G95 1) 2	G95 2	Umdrehungsvorschub in [mm/U, inch/U]
Gruppe 6		
G20 1) 1	G20 1) (G70) 1	Eingabesystem inch
G21 2	G21 (G71) 2	Eingabesystem metrisch
Gruppe 7		
G40 1) 1	G40 1) 1	Abwahl Fräserradiuskorrektur
G41 2	G41 2	Korrektur links von der Kontur
G42 3	G42 3	Korrektur rechts von der Kontur
Gruppe 8		
	G43 1	Werkzeuglängenkorrektur positiv ein
	G44 2	Werkzeuglängenkorrektur negativ ein
	G49 1) 3	Werkzeuglängenkorrektur aus
Gruppe 9		
G22 1		Arbeitsfeldbegrenzung, Schutzbereich 3 ein
G23 1) 2		Arbeitsfeldbegrenzung, Schutzbereich 3 aus
	G73 1	Tieflochbohrzyklus mit Späne brechen
	G74 2	Gewindebohrzyklus links
	G76 3	Feinbohrzyklus
	G80 1) 4	Zyklus aus
	G81 5	Bohrzyklus Plansenken
	G82 6	Bohrzyklus Ansenken
	G83 7	Tieflochbohrzyklus mit Späne entfernen
	G84 8	Gewindebohrzyklus rechts
	G85 9	Bohrzyklus
	G86 10	Bohrzyklus, Rückzug mit G00
	G87 11	Rückwärtssenken
	G89 13	Bohrzyklus, Rückzug mit Arbeitsvorschub
Gruppe 10		
G80 1) 1		Bohrzyklus aus
G83 2		Stirnflächentieflochbohren
G84 3		Stirnflächengewindebohren
G85 4		Stirnflächen-Bohrzyklus

Tabelle 2-2 Standardeinstellung ist mit 1) gekennzeichnet

ISO-Dialekt-T		ISO-Dialekt-M		Beschreibung
G87	5			Seitenflächentieflochbohren
G88	6			Seitenflächengewindebohren
G89	7			Seitenflächenbohren
		G98 1)	1	Rückkehr zum Ausgangspunkt bei Festzyklen
		G99	2	Rückkehr zum Punkt R bei Festzyklen
Gruppe 11				
G98 1)	1			Rückkehr zum Ausgangspunkt bei Bohrzyklen
G99	2			Rückkehr zum Punkt R bei Bohrzyklen
		G50 1)	1	Skalierung aus
		G51	2	Skalierung ein
Gruppe 12				
G66	1	G66	1	Makro-Modalaufruf
G67 1)	2	G67 1)	2	Makro-Modalaufruf löschen
Gruppe 13				
		G96	1	konstante Schnittgeschwindigkeit ein
		G97 1)	2	konstante Schnittgeschwindigkeit aus
Gruppe 14				
G54 1)	1	G54 1)	1	Nullpunktverschiebung anwählen
G55	2	G55	2	Nullpunktverschiebung anwählen
G56	3	G56	3	Nullpunktverschiebung anwählen
G57	4	G57	4	Nullpunktverschiebung anwählen
G58	5	G58	5	Nullpunktverschiebung anwählen
G59	6	G59	6	Nullpunktverschiebung anwählen
G54 P{1...48}	1	G54 P{1...48}	1	erweiterte Nullpunktverschiebungen
G54.1	7	G54.1	7	erweiterte Nullpunktverschiebung
G54 P0	1	G54 P0	1	“externe extOffset NV”
Gruppe 15				
		G61	1	Genauhalt modal
		G62	4	Automatischer Eck-Override
		G63	2	Gewindebohrmodus
		G64 1)	3	Bahnsteuerbetrieb
Gruppe 16				
G17	1			XY-Ebene
G18 1)	2			ZX-Ebene
G19	3			YZ-Ebene

Tabelle 2-2 Standardeinstellung ist mit 1) gekennzeichnet

ISO-Dialekt-T	ISO-Dialekt-M	Beschreibung
	G68 1	Rotation EIN 2D 3D
	G69 1) 2	Rotation AUS
Gruppe 17		
	G15 1) 1	Polarkoordinaten aus
	G16 2	Polarkoordinaten ein
Gruppe 18 (satzweise wirksam)		
G04 1	G04 1	Verweilzeit in [s] oder Spindelumdrehungen
G05 20	G05 18	high–speed cycle cutting
G05.1 22	G05.1 20	High speed cycle –> Aufruf CYCLE305
G07.1 18	G07.1 16	Zylinderinterpolation
	G08 12	Vorsteuerung EIN/AUS
	G09 2	Genauhalt
G10 2	G10 3	Nullpunktverschiebung/Werkzeugkorrektur schreiben
G10.6 19	G10.6 17	Schnellabheben EIN/AUS (T) Rückzug von der Kontur (POLF) (M)
	G11 4	Parametereingabe beenden
G27 16	G27 13	Referenzfahrcheck (in Vorbereitung)
G28 3	G28 5	1. Referenzpunkt anfahren
G30 4	G30 6	2./3./4. Referenzpunkt anfahren
G30.1 21	G30.1 19	Floating reference position
G31 5	G31 7	Messen mit schaltendem Taster
G52 6	G52 8	programmierbare Nullpunktverschiebung
G53 17	G53 9	Position im Maschinenkoordinatensystem anfahren
G65 7	G65 10	Makroaufruf
G70 8		Schlichtzyklus
G71 9		Abspanzyklus Längsachse
G72 10		Abspanzyklus Planachse
	G72.1 14	Konturwiederholung mit Rotation
	G72.2 15	Konturwiederholung linear
G73 11		Konturwiederholung
G74 12		Tieflochbohren und Einstechen in Längsachse (Z)
G75 13		Tieflochbohren und Einstechen in Planachse (X)
G76 14		Mehrfach-Gewindeschneidzyklus
G92 15	G92 11	Istwert setzen, Spndeldrehzahlbegrenzung
G92.1 23	G92.1 21	Istwert löschen, Rücksetzen des WKS

Tabelle 2-2 Standardeinstellung ist mit 1) gekennzeichnet

ISO-Dialekt-T	ISO-Dialekt-M	Beschreibung
Gruppe 20		
G50.2 1) 1		Polygondrehen AUS
G51.2 2		Polygondrehen EIN
Gruppe 21		
G13.1 1) 1		TRANSMIT AUS
G12.1 2		TRANSMIT EIN
Gruppe 22		
	G50.1 1	Spiegeln an programmierter Achse AUS
	G51.1 2	Spiegeln an programmierter Achse EIN
Gruppe 25		
	G13.1 1	Polarkoordinaten Interpolation AUS
	G12.1 2	Polarkoordinaten Interpolation EIN
Gruppe 31		
G290 1) 1	G290 1) 1	Anwahl Siemens-Mode
G291 2	G291 2	Anwahl ISO-Dialekt-Mode

Tabelle 2-3 G-Befehle sind im Siemens-Mode und im ISO-Dialekt-Mode funktional identisch

G-Befehle im Siemens-Mode	korrespondierende G-Befehle ISO-Dialekt-T	korrespondierende G-Befehle ISO-Dialekt-M
Gruppe 1: G00, G01, G02, G03, G33	Gruppe 1: G00, G01, G02, G03, G33	Gruppe 1: G00, G01, G02, G03, G33
Gruppe 6: G17, G18, G19	Gruppe 16: G17, G18, G19	Gruppe 2: G17, G18, G19
Gruppe 7: G40, G41, G42	Gruppe 7: G40, G41, G42	Gruppe 7: G40, G41, G42
Gruppe 8: G54 bis G554	Gruppe 14: G54 bis G59, G54P1 bis P48	Gruppe 14: G54 bis G59, G54 P1 bis P48
Gruppe 10: G60, G64		Gruppe15: G60, G64
Gruppe 13: G700, G710	Gruppe 6: G20, G21	Gruppe 6: G20, G21
Gruppe 14: G90, G91	Gruppe 3: G90, G91	Gruppe 3: G90, G91
Gruppe 15: G94 G95 G96 G961 G97 G971	Gruppe 5: G94 Gruppe 2: G97 Gruppe 5: G95 Gruppe 2: G97 Gruppe 5: G95 Gruppe 2: G96 Gruppe 5: G94 Gruppe 2: G96 Gruppe 5: G95 Gruppe 2: G97 Gruppe 5: G94 Gruppe 2: G97	Gruppe 5: G94 Gruppe 13: G97 Gruppe 5: G95 Gruppe 13: G97 Gruppe 5: G95 Gruppe 13: G96 Gruppe 5: G94 Gruppe 13: G96 Gruppe 5: G95 Gruppe 13: G97 Gruppe 5: G94 Gruppe 13: G97

Hinweis

Können einzelne G-Befehle der in Tabelle 2-3 genannten Gruppen nicht abgebildet werden, wird die in den Maschinendaten
20154: $MC_EXTERN_GCODE_RESET_VALUES bzw.
20152: $MC_GCODE_RESET_VALUES
abgelegte Grundstellung aktiviert.

Beispiel: `ISO-Mode`

```
N5   G00 X100. Y100.
N10  G90           ;aktiviert G90 im ISO-Mode Gruppe 3
                   ;im Siemens-Mode Gruppe 14
N15  G290          ;umschalten nach Siemens, G90 ist aktiv
N20  G91           ;aktiviert G91 im ISO-Mode Gruppe 3
                   ;im Siemens-Mode Gruppe 14
N25  G291          ;umschalten nach ISO-Mode
N30  G291          ;G91 ist aktiv
```

2.2.1 Anzeige des G-Codes

In der G-Code Anzeige werden immer die G-Codes der gerade aktiven Sprache angezeigt. Mit G290/G291 schaltet auch die G-Code Anzeige um.

Beispiel:

Mit einigen G-Befehlen vom ISO-Dialekt-Mode (z.B. G28) werden Siemens-Zyklen aufgerufen. Am Zyklenanfang wird DISPLOF programmiert, so daß für die Anzeige weiterhin die ISO–Dialekt-G-Befehle erhalten bleiben.

```
PROC  CYCLE328 SAVE DISPLOF
N10 ...
...
N99 RET
```

Ablauf:

- Externes Hauptprogramm ruft Siemens-Hüllzyklus auf. Beim Hüllzyklenaufruf wird implizit Siemens-Mode angewählt
- Durch DISPLOF wird die Satzanzeige auf den Aufruf-Satz eingefroren; die G-Code-Anzeige bleibt im externen Mode. Diese wird während der Bearbeitung des Siemens-Zyklus aktualisiert.

2.2.2 Anzeige nichtmodaler G–Codes

Ab SW 6.4 werden die externen nichtmodalen G–Codes (Gruppe 18) bei Satzwechsel nicht mehr zurückgesetzt, wenn diese G–Codes Unterprogramme aufrufen. Die G–Codes bleiben solange in der Anzeige sichtbar, bis aus diesem Unterprogramm zurückgesprungen wird.

Wird jedoch im Unterprogramm in den externen Sprachmodus gewechselt und ein anderer G–Code aus der Gruppe 18 programmiert, so wird der vorherige Wert überschrieben und der neue Wert bleibt bis zum Rücksprung stehen.

Beispiel:

Hauptprogramm		Anzeige Gruppe 18
N05 G00 X0 Y0	leer	
N08 G27 X10	–> ruft Cycle328	leer
N09 M0		leer
N40 M30		leer

Unterprogramm Cycle328	Anzeige Gruppe 18
N100 G290	G27
N102 X=$C_X	G27
N103 M0	G27
N104 G291	G27
N105 G30 X10 Y12 Z13	G30
N120 M99	G30

2.2.3 GCode–Ausgabe an PLC (ab SW 6.4)

In dem Maschinendatum 22 515: $MC_GCODE_GROUPS_TO_PLC_MODE wird das Verhalten der G–Gruppenübergabe an PLC beschrieben.

Beim bisherigen Verhalten ist die G–Gruppe der Array–Index eines 64 Byte großen Feldes (DBB 208 – DBB 271). Damit kann maximal die 64. G–Gruppe erreicht werden. Es können nur die G–Gruppen der Standard- oder der externen Sprache angezeigt werden.

Beim neuen Verhalten ist die Datenablage in der PLC maximal 8 Byte (DBB 208 – DBB 215 groß, d.h. es können insgesamt höchstens 8 G–Gruppen ausgegeben werden.

Bei diesem Verfahren ist der Array Index der Maschinendaten

22515: $MC_GCODE_GROUPS_TO_PLC[ ] bzw.
22512: $MC_EXTERN_GCODE_GROUPS_TO_PLC[ ]

gleich dem Array Index der Datenablage in der PLC (DBB 208 – DBB215).
Die GCode-Gruppe aus MD 22515: $MC_GCODE_GROUPS_TO_PLC[ ] wird in DBB 208 ausgegeben.

Vorteil ist, daß gleichzeitig GCodes des Siemens Mode und des ISO Mode ausge-

geben werden können.

Da in einem DBB2xx nur der G–Code einer Sprache ausgegeben werden kann, darf jeder Index (0 –7) nur bei einem der beiden Maschinendaten gesetzt werden, beim jeweils anderen MD muß der Wert 0 eingetragen sein. Fehler werden mit Alarm 4045 gemeldet.

Beispiel

$MC_GCODE_GROUPS_TO_PLC[0]=3
$MC_GCODE_GROUPS_TO_PLC[1]=0
$MC_GCODE_GROUPS_TO_PLC[2]=0
$MC_GCODE_GROUPS_TO_PLC[3]=0
$MC_GCODE_GROUPS_TO_PLC[4]=1
$MC_GCODE_GROUPS_TO_PLC[5]=2
$MC_GCODE_GROUPS_TO_PLC[6]=0
$MC_GCODE_GROUPS_TO_PLC[7]=0

$MC_EXTERN_GCODE_GROUPS_TO_PLC[0]=0
$MC_EXTERN_GCODE_GROUPS_TO_PLC[1]=3
$MC_EXTERN_GCODE_GROUPS_TO_PLC[2]=18
$MC_EXTERN_GCODE_GROUPS_TO_PLC[3]=1
$MC_EXTERN_GCODE_GROUPS_TO_PLC[4]=0
$MC_EXTERN_GCODE_GROUPS_TO_PLC[5]=0
$MC_EXTERN_GCODE_GROUPS_TO_PLC[6]=6
$MC_EXTERN_GCODE_GROUPS_TO_PLC[7]=31

an der PLC stehen dann folgende G–Codes zur Verfügung

DBB 208 = Gruppe 03 Siemens
DBB 209 = Gruppe 03 ISO–Dialekt
DBB 210 = Gruppe 18 ISO–Dialekt
DBB 211 = Gruppe 01 ISO–Dialekt
DBB 212 = Gruppe 01 Siemens
DBB 213 = Gruppe 02 Siemens
DBB 214 = Gruppe 06 ISO–Dialekt
DBB 215 = Gruppe 31 ISO–Dialekt

Beispiel für fehlerhafte Projektierung:

$MC_GCODE_GROUPS_TO_PLC[0]=3
$MC_GCODE_GROUPS_TO_PLC[1]=0
$MC_GCODE_GROUPS_TO_PLC[2]=0

$MC_EXTERN_GCODE_GROUPS_TO_PLC[0]=3 –>
Alarm 4045, Kanal K1 Konflikt zwischen Maschinendatum
{S$MC_GCODE_GROUPS_TO_PLC} und Maschinendatum
{S$MC_EXTERN_GCODE_GROUPS_TO_PLC}
$MC_EXTERN_GCODE_GROUPS_TO_PLC[1]=0
$MC_EXTERN_GCODE_GROUPS_TO_PLC[2]=18

Mit diesem Verfahren ist es nun möglich, gleichzeitig G–Codes des Standardmodus und des ISO–Dialekt_Modus anzuzeigen.

2.2.4 Nullpunktverschiebung

Bild 2-1 zeigt die Nullpunktverschiebungen (NV) des Siemens-Modes.

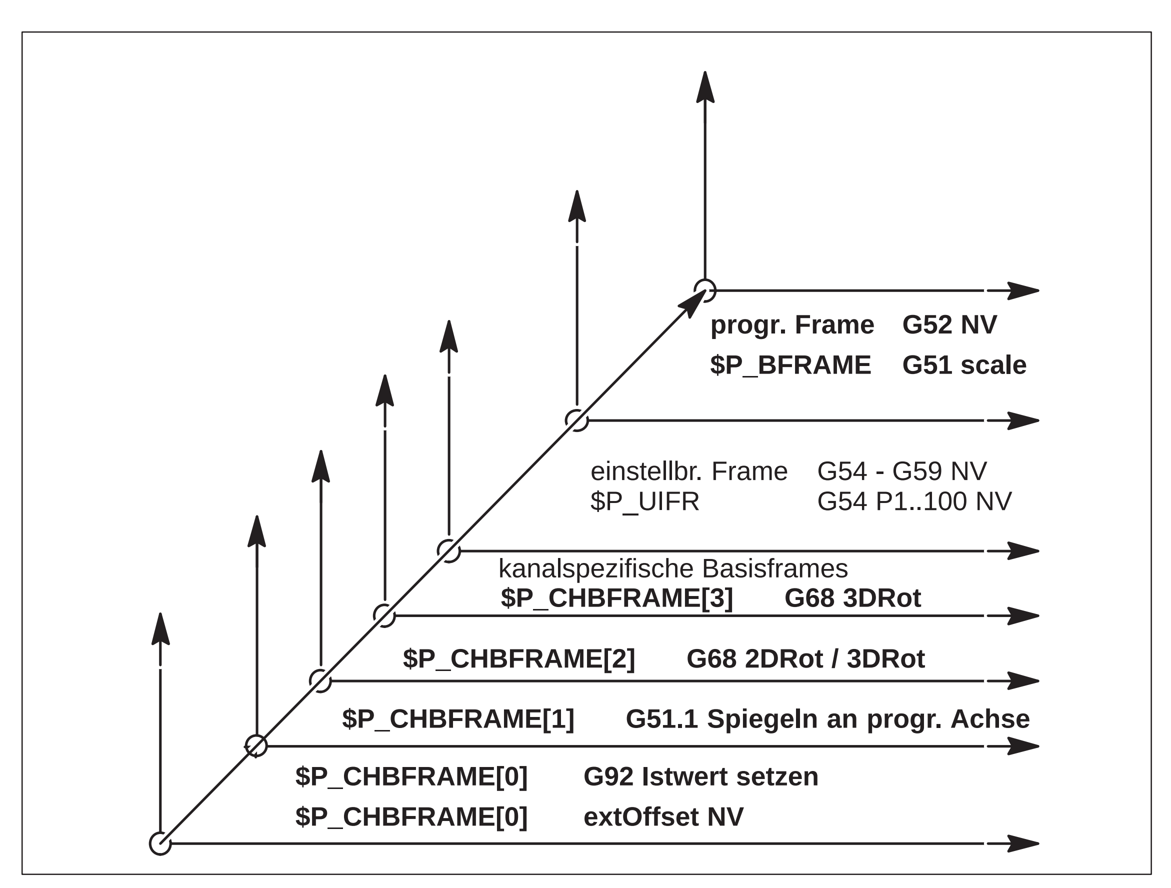


Bild 2-1 Momentane Abbildung der ISO Funktionen auf die Siemens Frames

Die im ISO-Mode vorhandenen Nullpunktverschiebungen werden auf die bestehenden Siemens Frames abgebildet. **Es gibt für den ISO-Mode keine eigene Frames.** Aktive Nullpunktverschiebungen werden in beiden Sprachmodes eingerechnet.
Änderungen im ISO-Mode wirken sich direkt im Siemens-Mode aus und umgekehrt.

Die Nullpunktverschiebungen gibt es sowohl im ISO–Dialekt–T wie im ISO–Dialekt–M:

- G52 ist eine programmierbare additive NV, wirksam bis Programmende oder Reset
- G54 bis G59 sind einstellbare Nullpunktverschiebungen
- G54 P1...P100 sind zusätzlich einstellbare Nullpunktverschiebungen
- G54 P0 ist eine “externe NV” extOffset

G54.1

G54.1 Pxx ist als alternative Schreibweise zu G54 Pxx realisiert. Die Funktionalität ist identisch. Bei G54.1 muß immer die Adresse P im Satz programmiert werden. Ist P nicht programmiert, wird der Alarm 12080 (Syntax-Fehler) ausgegeben.

Anzeige der erweiterten Nullpunktverschiebung G54 Pxx

Im ISO–Dialekt–T war es bisher nicht möglich, G54.1 P.. zu programmieren. Die G–Code–Gruppe 14 im ISO–Dialekt–T wird nun um das G–Code G54.1 erweitert und G54.1 wird jetzt bei programmiertem P standardmäßig angezeigt.

Bei Programmierung von G54 Pxx oder G54.1 Pxx wird bisher im ISO–Dialekt–M in der G–Code–Anzeige G54.1 angezeigt.

Über das MD $MC_EXTERN_FUNCTION_MASK Bit 11 kann nun aktiviert werden, daß in der G–Code–Anzeige auch das programmierte P nach dem Punkt angezeigt wird.

programmiert	Bitt 11 = 1	Bit 11 = 0
G54 P1	Anzeige G54P1	G54.1
G54 P28	Anzeige G54P28	G54.1
G54.1 P28	Anzeige G54P28	G54.1
G54 P48	Anzeige G54P48	G54.1
G54.1 P48	Anzeige G54P48	G54.1

2.2.5 Nullpunktverschiebung schreiben mit G10

Mit G10 können die Nullpunktverschiebungen aus dem Teileprogramm herausgeschrieben werden.

G10 L2 P1...P6 X.. Y..	;G54.. G59
G10 L20 P1...P100	;zusätzliche einstellbare NV
G10 L2 P0	externe extOffset NV

Diese Nullpunktverschiebungen werden auf die gleichen Frames abgebildet wie die bereits bei ISO–Dialekt–M bestehenden Nullpunktverschiebungen.

Der G10 Befehl wird für **ISO-Dialekt-T** erweitert:

Schreiben von Systemdaten
G10 Pxx X Y Z ;schreiben von Werkzeugkorrekturdaten

Abhängig von dem Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK, Bit1 wird mit G10 Pxx Werkzeuggeometrie oder Werkzeugverschleiß geschrieben.

$MC_EXTERN_FUNCTION_MASK, Bit1 = 0:
P > 100 Schreiben von Geometriewerten
P < 100 Schreiben von Verschleiwerten

$MC_EXTERN_FUNCTION_MASK, Bit 1=1:
P > 10000 Schreiben von Geometriewerten
P < 10000 Schreiben von Verschleiwerten

2.2.6 Dezimalpunkt Programmierung

Im ISO-Dialekt-Mode gibt es zwei Notationen für die Bewertung von programmierten Werten ohne Dezimalpunkt:

- **pocket calculator type notation**
  Werte ohne Dezimalpunkt werden als mm, inch oder grad interpretiert.
- **standard notation**
  Werte ohne Dezimalpunkt werden mit einem Umrechnungsfaktor multipliziert.

Die Einstellung erfolgt über das MD 10884, siehe Kap. 4 ”Inbetriebnahme”.

Es gibt zwei unterschiedliche Umrechnungsfaktoren, **IS-B** und **IS-C.** Diese Bewertung bezieht sich auf die Adressen X Y Z U V W A B C I J K Q R und F.

Beispiel Linearachse in mm:

X 100.5	entspricht Wert mit Dezimalpunkt: 100.5 mm	
X 1000	pocket calculator type notation: 1000 mm	
	standard notation:	IS-B: 1000* 0.001= 1 mm
		IS-C: 1000* 0.0001 = 0.1 mm

ISO-Dialekt Milling

Tabelle 2-4 Unterschiedliche Umrechnungsfaktoren für IS-B und IS-C

Adresse	Einheit	IS-B	IS-C
Linearachse	mm inch	0.001 0.0001	0.0001 0.00001
Rundachse	deg	0.001	0.0001
F Vorschub G94 (mm/inch pro min.)	mm inch	1 0.01	1 0.01
F Vorschub G95 (mm/inch pro min.)	mm inch	0.01 0.0001	0.01 0.0001
F Gewindesteigung	mm inch	0.01 0.0001	0.01 0.0001
C Fase	mm inch	0.001 0.0001	0.0001 0.00001
R Radius, G10 toolcorr	mm inch	0.001 0.0001	0.0001 0.00001
Q	mm inch	0.001 0.0001	0.0001 0.00001
I, J, K IpoParameter	mm inch	0.001 0.0001	0.0001 0.00001
G04 X oder U	s	0.001	0.001
A Winkel Konturzug	deg	0.001	0.0001
G74, G84 Gewindebohrzyklen $MC_EXTERN_FUNCTION_MASK Bit8 = 0 F als Vorschub wie G94, G95 Bit8 = 1 F als Gewindesteigung			

ISO-Dialekt Turning

Tabelle 2-5 Unterschiedliche Umrechnungsfaktoren für IS-B und IS-C

Adresse	Einheit	IS-B	IS-C
Linearachse	mm inch	0.001 0.0001	0.0001 0.00001
Rundachse	deg	0.001	0.0001
F Vorschub G94 (mm/inch pro min.)	mm inch	1 0.01	1 0.01
F Vorschub G95 (mm/inch pro Umdr.) $MC_EXTERN_FUNCTION_MASK			
Bit8 = 0	mm inch	0.01 0.0001	0.01 0.0001
Bit8 = 1	mm inch	0.0001 0.000001	0.0001 0.000001
F Gewindesteigung	mm inch	0.0001 0.000001	0.0001 0.000001
C Fase	mm inch	0.001 0.0001	0.0001 0.00001
R Radius, G10 toolcorr	mm inch	0.001 0.0001	0.0001 0.00001

Tabelle 2-5 Unterschiedliche Umrechnungsfaktoren für IS-B und IS-C

Adresse	Einheit	IS-B	IS-C
I, J, K IpoParameter	mm inch	0.001 0.0001	0.0001 0.00001
G04 X oder U		0.001	0.001
A Winkel Konturzug		0.001	0.0001
G76, G78 Gewindebohrzyklen $MC_EXTERN_FUNCTION_MASK Bit8 = 0 F als Vorschub wie G94, G95 Bit8 = 1 F als Gewindesteigung			
G84, G88 Gewindebohrzyklen $MC_EXTERN_FUNCTION_MASK Bit9 = 0 G95 F Bit8 = 1 G95 F	 mm inch mm inch	 0.01 0.0001 0.0001 0.000001	 0.01 0.0001 0.0001 0.000001

2.2.7 Schnellabheben mit G10.6

Mit G10.6 <Achsposition> kann eine Rückzugsposition für das Schnellabheben eines Werkzeugs (z.B. bei Werkzeugbruch) aktiviert werden. Die Rückzugsbewegung selbst wird mit einem digitalen Signal gestartet. Als Startsignal wird der 2. schnelle Eingang der NC verwendet.
Mit dem Maschinendatum $MN_EXTERN_INTERRUPT_NUM_RETRAC kann auch ein anderer schneller Eingang (1 – 8) ausgewählt werden.

Die Aktivierung der Rückzugsbewegung besteht im Siemensmode aus mehreren Teileprogrammbefehlen.

Aus

```
N10 G10.6 X19.5 Y33.3
```

wird NCK–intern

```
N10 SETINT (2) PRIO=1 CYCLE3106 LIFTFAST    ; Interrupteingang aktivieren
N30 LFPOS                                   ; Abhebemodus selektieren
N40 POLF[X]=19.5  POLF[Y]=33.3              ; Abhebepositionen für x19.5
                                            ; und y33.3 programmieren
N70 POLFMASK(X, Y)                          ; Rückzug der x und y Achse
                                            ; aktiv schalten
```

Diese Teileprogrammbefehle werden mit G10.6 intern in einem Satz zusammengefaßt.

Für die Aktivierung eines Interrupteingangs (SETINT(2)) muß auch ein Interruptprogramm (ASUP) definiert sein. Ist keines programmiert, kann das Teileprogramm nicht fortgesetzt werden, da es nach der Rückzugsbewegung mit einem Reset–Alarm abgebrochen wird. Für den Schnellrückzug mit G10.6 wird immer das Interruptprogramm (ASUP) CYCLE3106.spf verwendet. Ist das Programm CYCLE3106.spf im Teileprogrammspeicher nicht vorhanden, wird in einem Teileprogrammsatz mit G10.6 der Alarm 14011 “Programm CYCLE3106 nicht vor-

handen oder nicht zur Bearbeitung freigegeben” ausgegeben.

Das Verhalten der Steuerung nach dem Schnellrückzug wird Im ASUP CYCLE3106.spf festgelegt. Sollen die Achsen und die Spindel nach dem Schnellrückzug gestoppt werden, müssen in CYCLE3106.spf M0 und M5 programmiert werden. Ist CYCLE3106.spf ein Dummy–Programm, das nur M17 enthält, wird nach dem Schnellrückzug das Teileprogramm ohne Unterbrechung fortgesetzt.

Ist mit der Programmierung G10.6 <Achsposition> der Schnellrückzug aktiviert, wird mit dem Wechsel des Eingangssignals des 2. schnellen NC–Eingangs von 0 nach 1 die aktuelle Bewegung abgebrochen und die im G10.6–Satz programmierte Position mit Eilgang angefahren. Dabei werden die Positionen wie im G10.6–Satz programmiert, absolut oder inkrementell angefahren.

Die Deaktivierung der Funktion erfolgt mit G10.6 (ohne Positionsangabe). Der Schnellrückzug durch das Eingangssignal des 2. schnellen NC–Eingangs ist gesperrt.

Siemens

Die Funktion Schnellabheben mit G10.6 kann zum Teil mit der Funktion POLF[<achsname>] = <Rückzugsposition> abgedeckt werden. Die Funktion führt ebenfalls einen Rückzug auf die programmierte Position aus. Die weitere Funktionalität von ISO–Dialekt–Original ist aber nicht enthalten. Kann der Unterbrechungspunkt nicht auf direktem Weg angefahren werden, müssen die Hindernisse manuell umfahren werden.

Literatur. /PGA/, Programmieranleitung Arbeitsvorbereitung,
Kapitel “Erweitertes Stillsetzen und Rückziehen”

Einschränkungen

Es kann nur eine Achse für den Schnellrückzug programmiert werden.

2.2.8 Mehrgängie Gewinde mit G33

Mit der Syntax G33 X.. Z.. F.. Q.. werden im ISO–Dialekt–T und –M–Mode mehrgängige Gewinde programmiert. Dabei ist

X.. Z..	=	Endpunkt des Gewindes
F..	=	Gewindesteigung
Q..	=	Startwinkel

Gewinde mit versetzten Schnitten werden durch Angabe von zueinander versetzt liegenden Startpunkten im G33–Satz programmiert. Der Startpunktversatz wird unter der Adresse “Q” als absolute Winkelposition angegeben. Das zugehörige Settingdatum ($SD_THREAD_START_ANGLE) wird entsprechend verändert.

Beispiel: Q45000 Bedeutet: Startversatz 45,000 Grad

Wertebereich: 0.0000 bis 359.999 Grad

Der Startwinkel muß immer als Integerwert programmiert werden. Die Eingabefeinheit der Winkelangabe ist 0,001 Grad.

Beispiel:

```
N200 X50 Z80 G01 F.8 G95 S500 M3
N300 G33 Z40 F2 Q180000
```

Es wird ein Gewinde mit einer Steigung von 2mm und einen Startpunktversatz von 180 Grad hergestellt.

2.2.9 Gewinde mit variabler Steigung (G34)

Mit der Syntax G34 X.. Z.. F.. K.. werden im ISO–Dialekt–T und –M–Mode Gewinde mit variabler Steigung programmiert. Dabei bedeutet

X.. Z..	=	Endpunkt des Gewindes
F..	=	Gewindesteigung
K..	=	Gewindesteigungs–Zunahme (positiver Wert) bzw. Gewindesteigung–Abnahme (negativer Wert)

Mit G34 wird die Steigung bei jeder Spindelumdrehung um den unter der Adresse K programmierten Wert inkrementiert bzw. dekrementiert.

Beispiel:

```
N200 X50 Z80 G01 F.8 G95 S500 M3
N300 G91 G34 Z25.5 F2 K0.1
```

Der programmierte Weg von 25.5 mm entspricht 10 Spindelumdrehungen.

2.2.10 Verweilzeit in Spindelumdrehungen G04

Mit dem MD 20734: $MC_EXTERN_FUNCTION_MASK, Bit 2 wird bestimmt, wie die programmierte Verweilzeit in einem G04 Satz interpretiert wird. Die Verweilzeit kann mit G04 X U oder P programmiert werden.

Bit 2 = 0: Verweilzeit wird immer in [s] interpretiert.
Bit 2 = 1: Verweilzeit wird, wenn G95 aktiv ist, in Spindelumdrehungen interpretiert.

Im Fall von Standardnotation werden X und U Werte ohne Dezimalpunkt in interne Einheiten abhängig von IS-B bzw. IS-C umgerechnet. P wird immer in internen Einheiten interpretiert.

Beispiel:

```
N5 G95 G04 X1000
```

Standardnotation1000 * 0.001 = 1 Spindelumdrehung
pocketcalculator notation: 1000 Spindelumdrehungen

2.2.11 Skalieren und Spiegeln: G51, G51.1 (ISO–Dialekt–M)

Die Anwahl zum Skalieren und Spiegeln erfolgt mit G51, G51.1.
Beim Skalieren unterscheidet man zwei Möglichkeiten:

- achsiales Skalieren mit den Parametern I, J, K

Wird I, J, K nicht im G51 Satz programmiert, so wirkt der jeweilige Defaultwert aus den Settingdaten.
Negative achsiale Skalierungsfaktoren führen zusätzlich zum Spiegeln.

- skalieren in allen Achsen mit dem Skalierfaktor P

Wird kein P im G51 Satz geschrieben, wirkt der Defaultwert aus den Settingdaten.
Negative P-Werte sind nicht möglich.
Die Skalierungsfaktoren werden entweder mit 0,001 oder 0,00001 multipliziert.

Hinweis

Wird für die Parameter I, J, K ein anderer Faktor als ”1” programmiert oder fehlt die Adresse (Standardwert für I, J, K ist wirksam), wird die Kontur zusätzlich skaliert.

Beispiel

```
00512 (Teileprogramm)
N10 G17 G90 G00 X0 Y0          Ausgangsposition anfahren
N30 G90 G01 G94 F6000
N32 M98 P0513                  1) Kontur wie im UP programmiert
N34 G51 X0. Y0. I-1000 J1000   2) Kontur um X spiegeln
```

```
N36 M98 P0513
N38 G51 X0. Y0. I-1000 J-1000        3) Kontur um X und Y spiegeln
N40 M98 P0513
N42 G51 X0. Y0. I1000 J-1000         4) Kontur um Y spiegeln
N44 M98 P0513
N46 G50                              Abwahl Skalieren und Spiegeln
N50 G00 X0 Y0
N60 M30

O0513 (Unterprogramm)
N10 G90 X10. Y10.
N20 X50
N30 Y50
N40 X10. Y10.
N50 M99
```

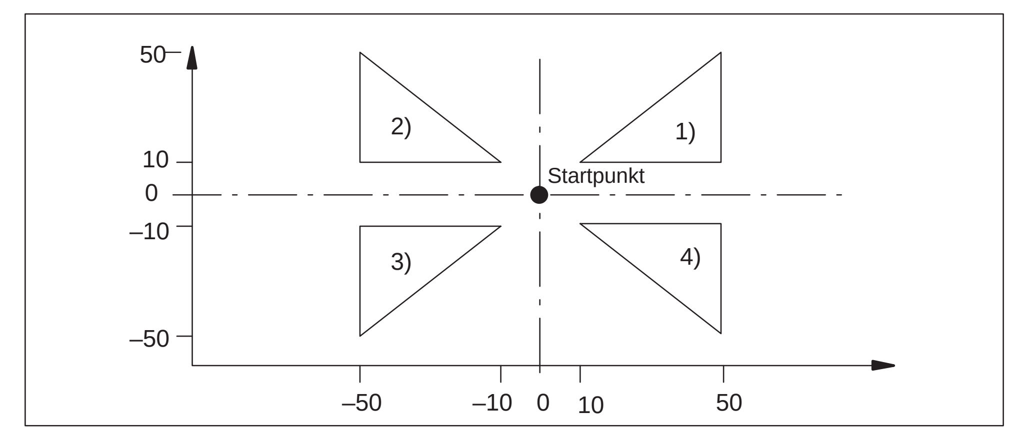


Bild 2-2 Skalieren und Spiegeln

Einstellungen der Systemparameter für das Beispiel Skalieren und Spiegeln:

MD 22910 $MC_WEIGHTING_FACTOR_FOR_SCALE = 0
MD 22914 $MC_AXES_SCALE_ENABLE = 1
MD 10884 $MN_EXTERN_FLOATINGPOINT_PROG = 0
MD 10886 $MN_EXTERN_INCREMENT_SYSTEM = 0

Bei MD $MC_AXES_SCALE_ENABLE = 0 ist achsiales Skalieren nicht möglich.

Beim Skalieren ist der Bezugspunkt immer der Werkstücknullpunkt, eine Programmierung vom Bezugspunkt ist nicht möglich.

Spiegeln

Mit G51.1 wird Spiegeln angewählt.
Gespiegelt wird an einer Spiegelachse, die parallel zu X,Y oder Z liegt und deren Position mit X, Y oder Z programmiert wird. Mit G51.1 X0 wird an der X–Achse gespiegelt, mit G51.1 X10 wird an einer Spiegelachse, die 10 mm parallel zur X–Achse verläuft gespiegelt.

Es können alle Achsen im Kanal, nicht nur die Geometrieachsen, gespiegelt wer-

den.
G51.1 wirkt additiv, d.h. nach N5 G51.1 X10 und N10 G51.1 Y10 ist Spiegeln in X und V aktiv.

Beispiel G51.1 X80.
An einer Spiegelachse, die parallel zu Y liegt und die X–Achse an der Position 80 schneidet, wird gespiegelt.

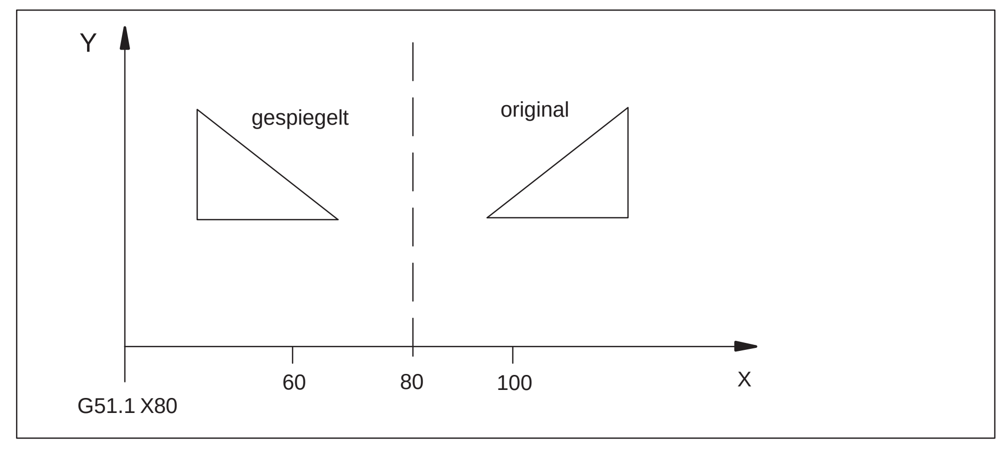


Bild 2-3 Spiegeln an einer zu Y parallelen Spiegelachse

Ist die Standardnotation aktiv (siehe Kapitel 2.2.6), werden die Achspositionen ohne Dezimalpunkt in internen Einheiten interpretiert.

Abgewählt wird Spiegeln mit G50.1 X0 Y0. Es kann auch achsweise abgewählt werden. Nach G50.1 X0 ist nur die Spiegelung in der X–Achse abgewählt, alle Spiegelungen in anderen Achsen bleiben aktiv.

G51.1 und G50.1 stehen alleine im Satz.
Abgebildet wird G51.1 in den kanalspezifischen Basisframe[1]. Dafürmuß das MD 28081: $MC_MM_NUM_BASE_FRAMES >=2 gesetzt sein.
Wenn im Siemensmode der Basisframe[1] verändert wird, hat das im ISO–Mode direkte Auswirkungen auf die Funktion.
Wird der Frame in allen Framekomponenten gelöscht, entspricht das einem G50.1 X0 Y0.. in allen Achsen.

G51.1 wird mit Reset abgewählt.

2.2.12 2D/3D Rotation G68 / G69 (ISO–Dialekt–M)

2D Rotation

Es wird das Koordinatensystem um die senkrechte Achse der angewählten Ebene gedreht.

Programmierung

G68 X.. Y.. R..

X.. Y..: Koordinaten des Drehpunktes, bezogen auf den aktuellen Werkstücknullpunkt. Ist keine Koordinate programmiert, liegt der Drehpunkt im Istwert. Der Wert wird immer absolut inter–pretiert.

R: Der Drehwinkel, abhängig von G90/G91, wirkt absolut oder inkrementell. Ist kein Winkel programmiert, wird der Win–kel aus dem Settingdatum 42150: $SA_DEFAULT_ROT_FACTOR_R aktiv. G68 muß alleine im Satz stehen.

G69 Rotation AUS; in diesem Satz können weitere Befehle pro–grammiert sein.

Abgebildet wird G68 auf den kanalspezifischen Basisframe 2. Dafür muß das MD 28081: $MC_MM_NUM_BASE_FRAMES >= 3 gesetzt sein.

Ein programmierter Winkel R wird nicht in das Settingdatum 42150: $SA_DEFAULT_ROT_FACTOR_R eingetragen. Dieses Settingdatum kann nur von Hand geschrieben werden und wirkt dann, wenn kein R im G68 Satz pro-grammiert ist.

3D Rotation

Der GCode G68 wird für 3D Rotation erweitert.

Programmierung

G68 X.. Y.. Z.. I.. J.. K.. R..

X.. Y.. Z..: Koordinaten des Drehpunktes, bezogen auf den aktuellen Werkstücknullpunkt. Ist keine Koordinate programmiert liegt der Drehpunkt im Werkstücknullpunkt. Der Wert wird immer absolut interpretiert. Die Koordinaten des Drehpunktes wir–ken wie eine Nullpunktverschiebung. G90/91 im Satz hat auf den G68 Befehl keinen Einfluß.

I.. J.. K..: Vektor im Drehpunkt. Das Koordiantensystem wird um die–sen Vektor mit dem Winkel R gedreht.

R: Drehwinkel, Der Drehwinkel wirkt immer absolut. Ist kein Winkel programmiert wird der Winkel aus dem Settingdatum

42150 $SA_DEFAULT_ROT_FACTOR_R aktiv.
G68 muß alleine im Satz stehen.

Die Unterscheidung 2D oder 3D Rotation, erfolgt nur über die Programmierung des Vektors I, J, K. Ist kein Vektor im Satz wird G68 2DRot angewählt. Ist ein Vektor im Satz wird G68 3DRot angewählt.

Ist ein Vektor mit der Länge 0 (I0, Y0, K0) programmiert, kommt es zu Alarm 12560 “Programmierter Wert außerhalb der zulässigen Grenzen“.

Mit G68 können 2 Drehungen hintereinander geschaltet werden. Ist in einem Satz mit G68 bisher kein G68 aktiv, wird die Rotation in den kanalspezifischen Basisframe 2 geschrieben. Ist G68 bereits aktiv, wird die Rotation in den kanalspezifischen Basisframe 3 geschrieben. Damit wirken beide Rotationen hintereinander.

Mit G69 wird 3D Rotation beendet. Sind zwei Rotationen aktiv, werden beide mit G69 abgewählt. G69 muß nicht alleine im Satz stehen.

2.2.13 Doppelschlitten– oder Doppelrevolverbearbeitung G68 / G69

Mit der Funktion G68 / G69 kann die beidseitige Bearbeitung von Drehteilen gesteuert werden. Sowohl die Bearbeitung mit einem Doppelschlitten in zwei Kanälen, als auch die Bearbeitung in einem Kanal mit zwei Werkzeugen die in einem Abstand x fest miteinander verbunden sind.

Mit dem MD $MN_EXTERN_DOUBLE_TURRET_ON wird definiert, ob die Bearbeitung in den beiden Kanälen synchronisiert (= FALSE) oder wechselweise eines von zwei festverbundenen Werkzeugen zur Bearbeitung eingesetzt wird (= TRUE).

Bei fest miteinander verbundenen Werkzeugen wird mit G68 der im MD 42162: $SC_EXTERN_DOUBLE_TURRET_DIST eingetragene Abstand x als additive Nullpunktverschiebung in der X–Achse aktiviert. Da das zweite Werkzeug die gegenüberliegende Seite des Drehteils bearbeitet, wird mit G68 zusätzlich ein Spiegeln um die Z–Achse aktiviert (Richtungswechsel der X–Achse). Mit dem nächsten Satz mit Achsbewegungen werden die Nullpunkverschiebung und das Spiegeln für das 2. Werkzeug aktiv.

Mit G69 wird die Nullpunktverschiebung wieder rückgängig gemacht und die Bearbeitung mit dem 1. Werkzeug fortgesetzt.

G68 und G69 müssen alleine im Satz programmiert werden.

Bei der Längenkorrektur in der X–Achse für das zweite Werkzeug muß das Vorzeichen der Korrektur berücksichtigt werden. Das Vorzeichen muß so eingegeben werden, als wenn die X–Achse nicht gespiegelt wäre. Oder es müssen die Settingdaten $SC_MIRROR_TOOL_LENGTH (Werkzeuglängenkorrektur spiegeln) und $SC_MIRROR_TOOL_WEAR (Verschleißwerte der Werkzeuglängenkorrektur spiegeln) gesetzt werden.

Das Maschinendatum $MN_MIRROR_REF_AX muß entweder = 0 oder = 1 sein. Damit wird immer die X– oder 1. Achse gespiegelt.

Wird G68 programmiert und G68 ist bereits aktiv, wird die G–Funktion überlesen. Dasselbe gilt für G69.

Doppelrevolverkopf: $MN_EXTERN_DOUBLE_TURRET_ON = TRUE

Das folgende Beispiel zeigt die Bearbeitung mit zwei fest verbundenen Werkzeugen. Damit die Funktion wirksam wird, muß das Maschinendatum $MN_EXTERN_DOUBLE_TURRET_ON = TRUE gesetzt sein.

Ist das Settingdatum 42162: $SC_EXTERN_DOUBLE_TURRET_DIST = 0, wird der Alarm “12728 Abstand für Doppelrevolver nicht gesetzt” ausgegeben.

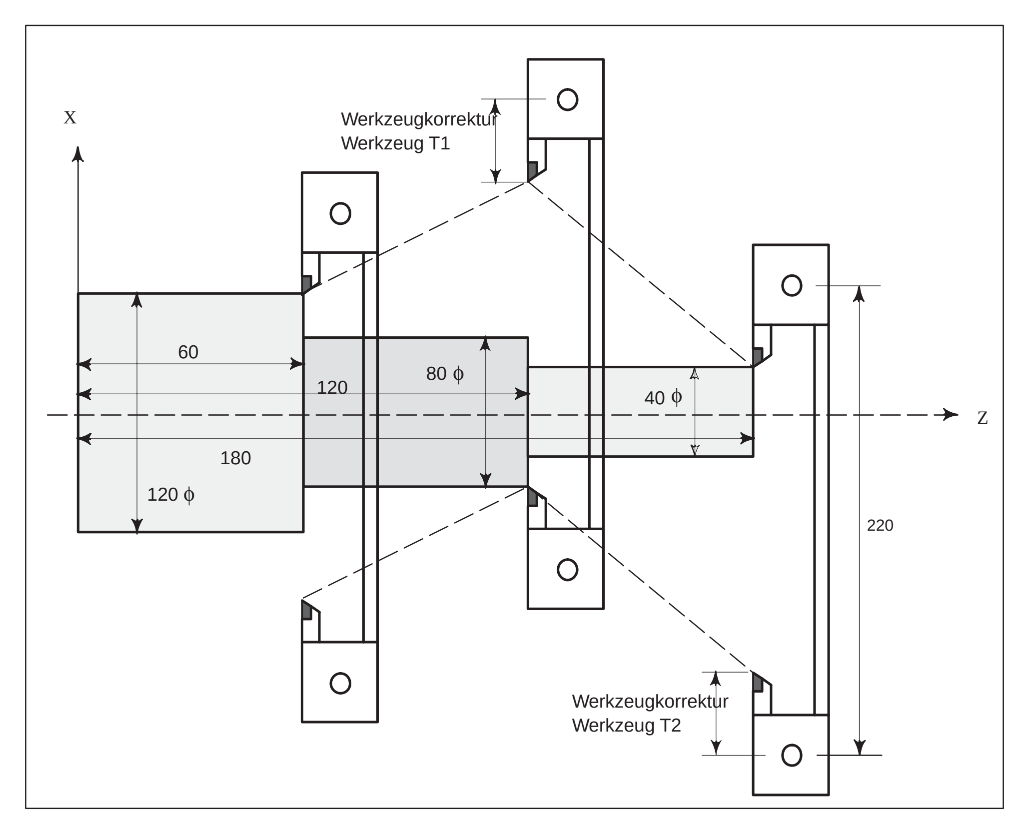


Bild 2-4 Bearbeitung mit 2 fest verbundenen Werkzeugen

Beispiel:

```
N100 X40. Z180. G1 F1 G95 S1000 M3 T1
N110 G68 ; spiegeln um Z und additive Nullpunktverschiebung (220mm) aktivieren
N120 X80. Z120. T2
N130 G69 ; spiegeln und Additive Nullpunktverschiebung deaktivieren
N140 X120. Z60 T1
```

Doppelschlittenbearbeitung: $MN_EXTERN_DOUBLE_TURRET_ON = FALSE

Ist das Maschinendatum $MN_EXTERN_DOUBLE_TURRET_ON = FALSE, wird mit G68 eine Kanalsynchronisation durchgeführt. Wird in einem Kanal G68 programmiert, wird die Bearbeitung so lange gestoppt, bis auch im zweiten Kanal G68 erkannt wird. Mit dieser Funktion kann der 1. und 2. Kanal synchronisiert werden. Eine weitere Synchronisarion wird nicht durchgeführt. Damit die beiden Werkzeuge im weiteren Verlauf der Bearbeitung synchron verfahren, müssen in beiden Kanälen die programmierten Bewegungen und Vorschübe gleich sein.

Zur Synchronisation der ersten beiden Kanäle wird für G68 die Wait–Marke 1 und für G69 die Wait–Marke 2 verwendet. Es dürfen deshalb nicht gleichzeitig die ersten beiden M–Funktionen für die Kanalsynchronisation im selben Teileprogramm verwendet werden (siehe Kap. 4.1.10) .

G68 wirkt nur in den ersten beiden Kanälen. Wird G68 in einem anderen Kanal programmiert und das Maschinendatum $MN_EXTERN_DOUBLE_TURRET_ON = FALSE, wird G68 überlesen.

Die Funktion wird zur Herstellung dünner Drehteile verwendet. Die beiden Werkzeuge sollen deshalb auf der jeweils gegenüberliegenden Seite des Drehteils die selbe Bewegung, um die Z–Achse gespiegelt, ausführen. Dafür müssen in beiden Kanälen die gleichen Verfahrbewegungen und Vorschübe programmiert werden.

Beispiel für die synchrone Bearbeitung mit zwei Kanälen.

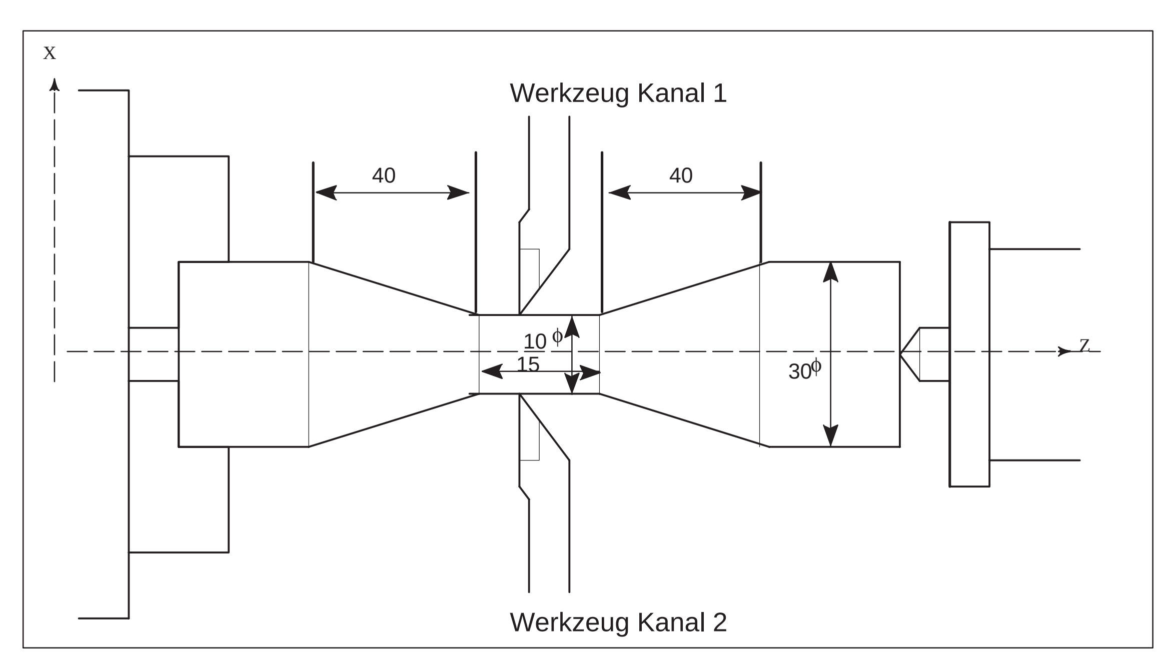


Bild 2-5 Synchrone Bearbeitung mit 2 Kanälen

Beispiel:
Kanal 1:

```
N10 ....
– “ –
N1000 G68                ; Synchronisation starten
N1010 G01 X30 Z120 G95 F1.2 S1000 M3
```

```
N1020 X15 Z80
N1030 Z65
N1040 Z25 X40
N1050 G69                ; Synchronisation aus
```

Kanal 2:

```
N10 .....
– " –
N2000 G68                ; Synchronisation starten
N2010 X30 Z120 G01 G95 F1.2 S1000 M3
N2020 X15 Z80
N2030 Z65
N2040 X40 Z25
N2050 G69                ; Synchronisation aus
```

Bei ISO–Dialekt–Original wird bei aktivem G68 ebenfals eine Kanalsynchronisation durchgeführt.

2.2.14 Polarkoordinaten: G15 (ISO-Dialekt-M)

Bei ISO-Dialekt-Mode müssen NC-Programmabschnitte, in denen polar programmiert wird, mit dem Start-Befehl G16 begonnen werden. Bis zum Ende-Befehl G15 werden die Koordinaten der Endpunkte als Polarkoordinatenwerte Radius und Winkel in der aktuellen Ebene interpretiert.

Die erste Achse der Ebene ist der Polradius, die zweite Achse der Polwinkel, d.h. bei G17 ist X der Radius, Y der Winkel.

Nach G16 wird in jedem Satz bis G15 der Pol neu gesetzt und zwar bei G17:

- G90 X der Pol liegt im Werkstücknullpunkt
- G91 X der Pol liegt in der aktuellen Position
- kein X im Satz der Pol liegt im Werkstücknullpunkt

Wird der Pol von der aktuellen Position auf den Werkstücknullpunkt gelegt, so wird der Radius als die Entfernung von der aktuellen Position zum Werkstücknullpunkt berechnet.

Beispiel:

```
N5 G17 G90 X0 Y0
N10 G16 X100. Y45.      Polarkoordinaten EIN, Pol ist Werkstücknullpunkt,
                        Position X 70,711 Y 70,711 im kartesischen Koor-
                        dinatensystem
N15 G91 X100 G90 Y0     Pol ist aktuelle Position, Position X 170,711 Y 70,711
N20 Y90.                kein X im Satz, Pol liegt im Werkstücknullpunkt,
                        Radius = SORT(X*X +Y*Y) = 184,776
```

Der Polradius wird immer absolut gefahren, der Polwinkel kann absolut oder inkrementell verfahren werden.

Programmierter Winkel

Bei aktiver Polarkoordinatenprogrammierung kann der programmierte Winkel über die Systemvariable $P_AP gelesen werden.
Eingesetzt wird diese Variable in den Hüllzyklus. Bevor der Pol neu gesetzt wird, muß bei inkrementeller Programmierung der Winkel gespeichert werden, da der Winkel gelöscht wird.

Mit G15 wird die Polprogrammierung beendet. Dabei wird der Polradius auf 0 gesetzt.

2.2.15 Polarkoordinaten Interpolation G12.1 / G13.1 (G112/G113)

Mit G12.1 und G13.1 wird eine Interpolation in der Bearbeitungsebene zwischen einer Rundachse und einer Linearachse ein- und ausgeschaltet. Eine weitere Linearachse steht senkrecht auf dieser Ebene. Diese Funktion entspricht der TRANSMIT–Funktion im Siemensmode. Im Siemensmode können zwei TRANSMIT-Transformationen parametriert werden. Für G12.1 wird immer der 1. TRANSMIT-Datensatz, der dem 2. Transformationsdatensatz entsprechen muß. Eine ausführliche Beschreibung der TRANSMI-Funktion ist der Dokumentation

/FB2/ SINUMERIK 840D/810D(CCU2)
Funktionsbeschreibung Erweiterungsfunktionen, Kapitel M1 und
/PGA/ SINUMERIK 840D/810D
Programmieranleitung Arbeitsvorbereitung, Kapitel ”Transformationen”
zu entnehmen

Beispiel

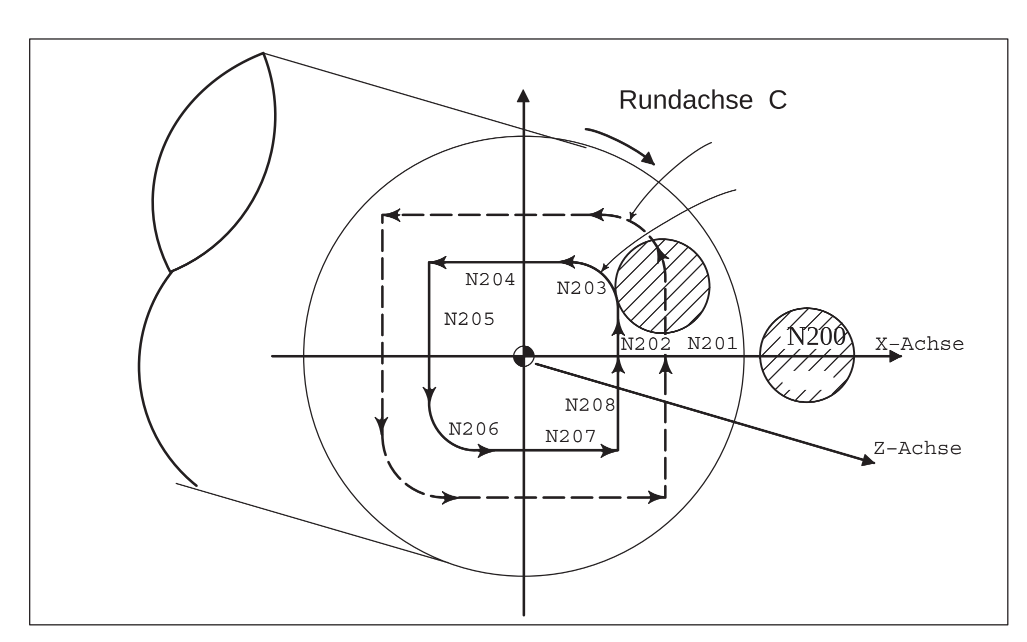


Bild 2-6 Beispiel für Polarkoordinaten Interpolation

```
00001
N010 T0101
N0100 G90 G00 X60.0 C0 Z..
N0200 G12.1                    ;TRANSMIT–Anwahl
N0201 G42 G01 X20.0 F1000
N0202 C10.0 ;
N0203 G03 X10.0 C20.0 R10.0
N0204 G01 X–20.0
N0205 C–10.0
N0206 G03 X–10.0 C–20.0 I10.0 J0
N0207 G01 X20.0
N0208 C0
N0209 G40 X60.0
N0210 G13.1                    ;TRANSMIT–Abwahl
N0300 Z..
N0400 X.. C..
N0900 M30
```

Hinweis

Es darf kein Geoachstausch (parallele Achsen mit G17 (g18, G19)) aktiv sein.

2.2.16 Zylinderinterpolation G07.1 (G107)

Mit der Funktion G07.1 (Zylinderinterpolation) können beliebig verlaufende Nuten an zylindrischen Körpern gefräst werden. Der Verlauf der Nuten wird bezogen auf die abgewickelte, ebene Zylindermantelfläche programmiert. Die Zylinderinterpolation wird mit G07.1 mit Angabe des Zylinderradius gestartet G07.1 C<Zylinderradius> und mit G07.1 C0 (Radius = 0) beendet.

Die Funktion wird intern auf die Siemensfunktionalität von TRACYL abgebildet. Im ISO-Dialekt-Mode wird mit G07.1 immer die 1. TRACYL–Transformation und der 1. Transformationsdatensatz aktiviert. Die 2. TRACYL–Funktion kann im ISO-Dialekt-Mode nicht aktiviert werden. Zur ausführlichen Beschreibung und Parametrierung der 1. TRACYL–Funktion ist der Dokumentation:

/FB2/ SINUMERIK 840D/810D(CCU2)
Funktionsbeschreibung Erweiterungsfunktionen, Kapitel M1 und
/PGA/ SINUMERIK 840D/810D
Programmieranleitung Arbeitsvorbereitung, Kapitel ”Transformationen” zu entnehmen

Einschränkungen

Bei Siemens muß die Rundachse für die Zylinderinterpolation in Maschinendaten definiert werden.
Bei ISO-Dialekt wird die Rundachse für die Zylinderinterpolation mit der Programmierung G07.1 <Rundachsname>... definiert.

Beispiel

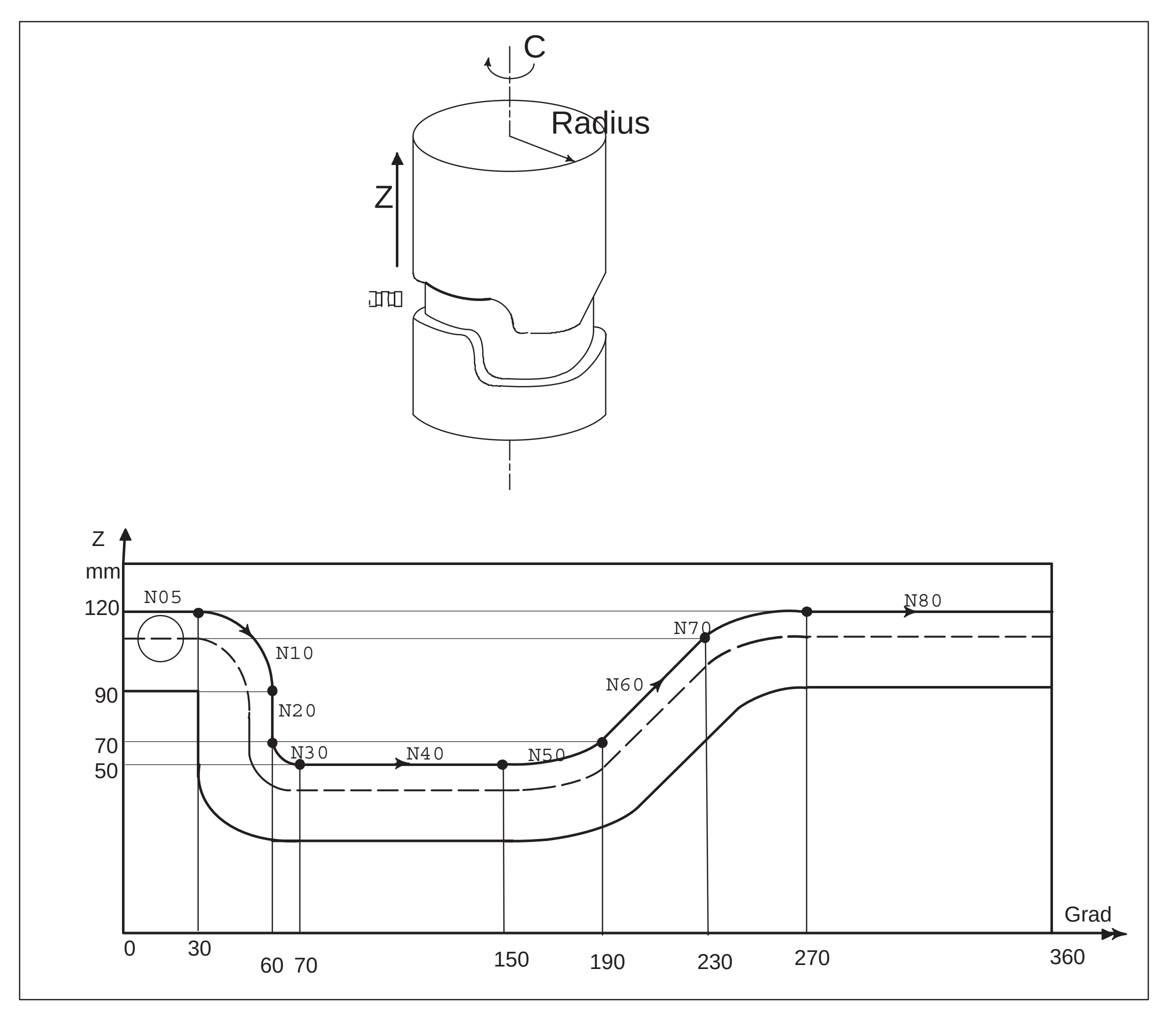


Bild 2-7 Beispiel für die Zylinderinterpolation G07.1

Programmierbeispiel im ISO-Dialekt-Mode:

```
%0001
N05 G00 G90 Z100.0 C0
N10 G01 G91 G18 Z0 C0
N20 G07.1 C57299             ;Zylinderinterpolation anwählen  mit Radius
                             ;  57.299 mm

N30 G90 G01 G42 Z120.0 D01 F250

N40 C30.0
```

```
N50 G02 Z90.0 C60.0 R30.0
N60 G01 Z70.0
N70 G03 Z60.0 C70.0 R10.0
N80 G01 C150.0
N90 G03 Z70.0 C190.0 R75.0
N100 G01 Z110.0 C230.0
N110 G02 Z120.0 C270.0 R75.0
N120 G01 C360.0
N130 G40 Z100.0
N140 G07.1 C0                ; Zylinderinterpolation abwählen
N150 M30 ;
```

Programmierbeispiel im Siemensmode: Die Y–Achse ist der Rundachse als Linearachse zugeordnet.

```
%0001
N05 G00 G90 Z100 C0
N10 G01 G91 G18 Z0 C0;
N20 TRACYL(114.598)          ;Zylinderinterpolation anwählen mit
                             ; Radius 57.299 mm
N30 G90 G01 G42 Z120 D01 F250
N40 Y30
N50 G02 Z90 Y60 RND=30
N60 G01 Z70
N70 G03 Z60.0 Y70 RND=10
N80 G01 Y150
N90 G03 Z70 Y190 RND=75
N100 G01 Z110 Y230
N110 G02 Z120 Y270 RND=75
N120 G01 Y360
N130 G40 Z100
N140 TRAFOOF                 ;Zylinderinterpolation abwählen
N150 M30 ;
```

2.2.17 Interrupt–Programm mit M96 / M97 (ASUP)

M96

Mit M96 P<Programmnummer> kann ein Unterprogramm als Interruptroutine definiert werden.

Der Start dieses Programms wird durch ein externes Signal ausgelöst. Für den Start der Interruptroutine wird von den 8 im Siemensmode zur Verfügung stehenden Eingängen immer der 1. schnelle NC–Eingang verwendet. Mit dem Maschinendatum 10818: $MN_EXTERN_INTERRUPT_NUM_ASUP kann auch ein anderer schneller Eingang (1 – 8) ausgewählt werden.

Die Funktion wird auf die Siemenssyntax**:** SETINT(x) <CYCLE396> [PRIO=1] ab-

gebildet.

Im Hüllzyklus CYCLE396 wird das mit Pxxxx programmierte Interruptprogramm im ISO–Mode aufgerufen. Die Programmnummer steht in $C_PI. Am Ende des Hüllzyklus wird das Maschinendatum
10808: $MN_EXTERN_INTERRUPT_BITS_M96 BIT1 ausgewertet und entweder mit REPOSA auf den Unterbrechungspunkt positioniert oder mit dem nächsten Satz fortgesetzt. In der neuen Zyklenvariablen $C_PI steht der mit ”P” programmierte Wert ohne führende Null. Im Hüllzyklus müssen diese vor dem Aufruf des Unterprogramms auf vier Stellen ergänzt werden.

Beispiel: N0020 M96 P5

Aufruf im Hüllzyklus
progName = ”000” << $C_PI
ISOCALLprogName

Siehe auch Behandlung von 8-stelligen Programmnummern, wenn MD 20734: $MC_EXTERN_FUNCTION_MASK, Bit 6 gesetzt ist.

M97

Mit M97 wird der Start der Interruptroutine unterdrückt. Erst nach der nächsten Aktivierung mit M96 kann die Interruptroutine mit dem externen Signal gestartet werden.

Das entspricht der Siemenssyntax**:** ENABLE(x).

x = Inhalt von MD 10818: $MN_EXTERN_INTERRUPT_NUM_ASUP

Soll das mit M96 Pxx programmierte Interruptprogramm mit dem Interruptsignal direkt aufgerufen werden (ohne Zwischenschritt mit CYCLE396), muß das Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK Bit 10 gesetzt werden. Das mit Pxx programmierte Unterprogramm wird dann bei einem Signalwechsel von 0 –> 1 im Siemensmode aufgerufen.

Die M–Funktionsnummern für die Interruptfunktion werden über Maschinendaten eingestellt. Mit dem Maschinendatum
10804 $MN_EXTERN_M_NO_SET_INT wird die M–Nummer zum Aktivieren einer Interruptroutine, mit dem Maschinendatum
10806 $MN_EXTERN_M_NO_DISABLE_INT wird die M–Nummer zum Unterdrükken einer Interruptroutine, bestimmt.

Es dürfen nur solche M–Funktionen eingestellt werden, die keine Standard–M–Funktionen sind. Die Voreinstellung der M–Funktionen ist M96 und M97. Um die Funktion zu aktivieren, muß im Maschinendatum
10808: $MN_EXTERN_INTERRUPT_BITS_M96, Bit 0 gesetzt werden. Die M–Funktionen werden dann nicht an die PLC ausgegeben. Ist das Bit 0 nicht gesetzt, werden die M–Funktionen als normale Hilfsfunktionen interpretiert.

Nach dem Ende des ”Interrupt”– Programms, wird standardmäßig auf die Endposition des auf den Unterbrechungssatz folgenden Teileprogrammsatzes gefahren. Soll das Teileprogramm vom Unterbrechungspunkt aus weiter bearbeitet werden, muß eine REPOS– Anweisung am Ende des ”Interrupt”– Programms stehen, z.B.

REPOSA. Dafür muß das Interruptprogramm im Siemensmode geschrieben sein.

Die M–Funktionen zum Aktivieren und Deaktivieren eines Interruptprogramms müssen alleine im Satz stehen. Werden weitere Adressen außer “M” und “P” im Satz programmiert, wird der Alarm 12080 (Syntaxfehler) ausgegeben.

Hinweis für Bearbeitungszyklen

Bei ISO-Dialekt-Original kann eingestellt werden, ob ein Bearbeitungszyklus sofort oder erst am Ende durch eine Interruptroutine unterbrochen werden soll. Die Hüllzyklen müssen dafür das Maschinendatum
10808: $MN_INTERRUPT_BITS_M96 Bit 3 auswerten. Ist das Bit=1 muß am Zyklenanfang der Interrupt mit DISABLE(1) gesperrt und am Zyklenende mit ENABLE(1) wieder aktiviert werden, damit der Bearbeitungszyklus nicht unterbrochen wird.
Da das Interrupt-Programm nur bei Flankenwechsel 0/1 gestartet wird, muß der Interrupteingang bei gesperrtem Interrupt während der Zyklenlaufzeit mit einer Synchronaktion im Hüllzyklus überwacht werden. Ist das Interruptsignal im Zyklus von 0 nach 1 gewechselt, muß am Ende des Hüllzyklus das Interruptsignal nach dem ENABLE(1) noch einmal gesetzt werden, damit anschließend das Interrupt-Programm gestartet wird. Um im Hüllzyklus den Interrupteingang schreiben zu können, muß das Maschinendatum
10361: $MN_FASTIO_DIG_SHORT_CIRCUIT[1] parametriert werden.

Maschinendaten

10808: MD $MN_EXTERN_INTERRUPT_BITS_M96:

Bit 0: = 0, kein Interrupt–Programm möglich, M96/M97 sind normale M–Funktionen
= 1, Aktivierung eines Interrupt–Programms mit M96/M97 erlaubt

Bit 1: = 0, Teileprogramm mit der Endposition des nächsten Satz nach dem Unterbrechungssatz weiterbearbeiten
= 1, Teileprogramm ab der Unterbrechungsposition weiterbearbeiten
(Auswertung im Interruptprogramm (ASUP), return mit/ohne REPOSL)

Bit 2: = 0, das Interruptsignal unterbricht den aktuellen Satz sofort und startet die Interruptroutine
= 1, die Interruptroutine wird erst am Ende des Satzes gestartet.

Bit 3: = 0, Bearbeitungszyklus bei einem Interruptsignal unterbrechen
= 1, Interrupt–Programm erst am Ende des Bearbeitungszyklus starten.
(Auswertung in den Hüllzyklen)

Das Bit 3 muss in den Hüllzyklen ausgewertet werden und der Zyklenablauf entsprechen angepasst werden.

Bit 1 muß im Interruptprogramm ausgewertet werden. Ist Bit 1 = TRUE, muß am Programmende mit REPOSL auf den Unterbrechungspunkt positioniert werden.

Beispiel:

```
N1000 M96 P1234     ;ASUP 1234.spf aktivieren. bei steigender Flanke des
                    ; 1. schnellen Eingangs, wird das Programm 1234.spf
                    ; gestartet
         "
         "
N3000 M97           ;Deaktivieren des ASUP
```

Vor dem Aufruf des Interruptprogramms wird kein Schnellabheben (LIFTFAST) ausgeführt. Mit der steigenden Flanke des Interruptsignals wird, abhängig von MD 10808: $MN_EXTERN_INTERRUPT_BITS_M96, das Interruptprogramm sofort gestartet.

Einschränkungen Siemens

Die Interruptroutine wird wie ein normales Unterprogramm behandelt. Das heißt, um die Interruptroutine ausführen zu können muß mindestens eine Unterprogrammebene frei sein. (Bei Siemens stehen 12 Programmebenen zur Verfügung, bei ISO–Dialekt sind es 5).

Die Interruptroutine wird nur bei einem Flankenwechsel des Interruptsignals von 0 nach 1 gestartet. Bleibt das Interruptsignal permanent auf 1 stehen, wird die Interruptroutine nicht mehr neu gestartet

Einschränkungen ISO-Dialekt

Für die Interruptroutine ist eine Programmebene reseviert, so daß alle zulässigen Programmebenen vor dem Aufruf des Interruptprogramms belegt sein dürfen.

Abhängig von Maschinendaten wird auch bei statisch anstehendem Signal das Interruptprogramm gestartet.

2.2.18 Kommentare

Im ISO-Dialekt-Mode werden runde Klammern als Kommentarzeichen interpretiert. Im Siemens-Mode wird “;” als Kommentar interpretiert. Zur Vereinfachung werden im ISO-Dialekt-Mode “;” ebenfalls als Kommentar verstanden.
Wird innerhalb eines Kommentars erneut das Kommentaranfangszeichen verwendet ’(’, dann wird der Kommentar erst beendet, wenn alle offenen Klammern wieder geschlossen sind.

Beispiel:

```
N5 (Kommentar) X100 Y100
N10 (Kommentar(Kommentar)) X100 Y100
N15 (Kommentar(Kommentar) X100) Y100
```

Im Satz N5 und N10 wird X100 Y100 ausgeführt, im Satz N15 nur Y100, da die erste Klammer erst nach X100 geschlossen wird. Bis dahin wird alles als Kommentar interpretiert.

2.2.19 Block skip

Das Skipzeichen ”/” kann an jeder beliebigen Stelle im Satz, also auch mitten im Satz stehen. Ist die programmierte Skipebene zum Zeitpunkt des Übersetzens aktiv, wird der Satz von dieser Stelle an bis zum Satzende nicht übersetzt. Eine aktive Skipebene bewirkt also das gleiche wie ein Satzende.

Beispiel:

```
N5 G00 X100. /3 YY100        —> Alarm 12080,
N5 G00 X100. /3 YY100        —>  kein Alarm, wenn Skipebene 3 aktiv ist
```

Skipzeichen innerhalb eines Kommentares werden nicht als Skipzeichen interpretiert

Beispiel:

```
N5 G00 X100. ( /3 Teil1 ) Y100  ;auch bei aktiver Skipebene 3 wird die
                                ;Y-Achse verfahren
```

Es kann die Skipebene /1 bis /9 werden. Skipwerte <1 >9 führen zu Alarm 14060
Abgebildet wird die Funktion auf die bestehenden Siemens Ausblendebenen. Im Gegensatz zu ISO-Dialekt-Original sind / und /1 getrennte Ausblendebenen, die auch getrennt aktiviert werden müssen.

2.2.20 Hilfsfunktionsausgabe

M

ISO-Dialekt-Mode

M-Funktionen werden als Hilfsfunktionen an die PLC ausgegeben. Ausgenommen davon sind nur M98 und M99. Alle anderen vordefinierten M-Funktionen werden als Hilfsfunktionen an die PLC gegeben.
Vordefinierte M-Funktionen sind:
M17, M40, M41, M42, M43, M44, M45, M70, M96, M97, M98, M99.

Spindel-Achs-Umschaltung mit M29

Im ISO-Dialekt-Mode wird die Spindel mit M29 in den Achsbetrieb geschaltet. Die M-Funktionsnummer kann aber auch über ein Maschinendatum variabel eingestellt werden.
Zur Voreinstellung der M-Funktionsnummer wird das MD 20095 $MC_EXTERN_RIGID_TAPPING_M_NR verwendet. Das Maschinendatum ist nur im externen Sprachmode wirksam und ist mit M29 voreingestellt. Es darf nur mit solchen M-Funktionsnummern belegt werden, die nicht als Standard M-Funktionen verwendet werden. Nicht erlaubt sind die M-Funktionsnummern M0-M5, M30, M98, M99.
Die selbe Funktion wird im Siemens-Mode mit M70 ausgeführt.
Zur Voreinstellung der M-Funktionsnummer wird das MD 20094 $MC_SPIND_RIGID_TAPPING_M_NR) verwendet. Das Maschinendatum ist nur im Siemens-Mode wirksam und ist mit M70 voreingestellt. Damit kann auch im Siemens-Mode für die Spindelumschaltung eine andere M-Funktion als M70 eingestellt werden. Das Maschinendatum darf nur mit solchen M-Funktionsnummern belegt sein, die nicht als Standard M-Funktionen verwendet werden. Nicht erlaubt sind M0-M5, M17, M19, M30, M40-M45.

H

Alle H-Funktionen werden bei ISO-Dialekt-M als Hilfsfunktion an die PLC ausgegeben. Bei ISO-Dialekt-T G Code System A ist H - nur beim Vorhandensein einer 4. Achse - der inkrementelle Weg der 4.Achse.

T

T-Funktionen werden als Hilfsfunktionen an die PLC ausgegeben. Darüber hinaus hat T die Bedeutung einer Werkzeuganwahl.

D

Die D-Funktion wird als Hilfsfunktion an die PLC ausgegeben. Bei ISO-Dialekt-M wird unter der Adresse D die Werkzeuglängenkorrektur aktiviert.

B

Ist B keine Achse, wird die B-Funktion als Hilfsfunktion H mit Adresserweiterung H1= an die PLC ausgegeben.
Beispiel: B1234 wird als H1 = 1234 ausgegeben.

2.2.21 1. Referenzpunkt anfahren: G28

Bei Einlesen des ISO-Dialekt-Befehls ”G28 <Achse>” wird automatisch der Zyklus CYCLE328 aufgerufen. Unter <Achse> wird die Zwischenposition (inkrementell oder absolut) angegeben, über die der Referenzpunkt angefahren werden soll. Die Zwischenposition und die Referenzposition werden im Positioniermodus angefahren.

Der Zyklus gilt nur für die bei ISO-Dialekt möglichen Achsen:

- ISO-Dialekt-M: X, Y, Z (A, B, C, U, V, W)
- ISO-Dialekt-T: X, Z, Y (C)

Der Zyklus wird immer mit Radiusprogrammierung (DIAMOF) ausgeführt. Nach Beendigung des Zyklus sind die vor Aufruf des Zyklus aktiven G-Befehle wieder wirksam.

Vor dem Anfahren des 1. Referenzpunktes müssen verschiedene Maschinendaten gesetzt sein, siehe Kap. 4 ”Inbetriebnahme”.

2.2.22 Vorsteuerung Ein-/Ausschalten mit G08 P..

Durch die Vorsteuerung wird der geschwindigkeitsabhängige Nachlaufweg beim Bahnfahren gegen Null reduziert. Fahren mit Vorsteuerung ermöglicht höhere Bahngenauigkeit und damit bessere Fertigungsergebnisse.

Hinweis

Über Maschinendaten wird die Art der Vorsteuerung festgelegt und welche Bahnachsen vorgesteuert verfahren werden sollen.

Standard: Geschwindigkeitsabhängige Vorsteuerung.
Option: Beschleunigungsabhängige Vorsteuerung.

Beispiel

```
N0010 G08 P1                ; Vorsteuerung einschalten
N0020 G1 X10  Y50  F900
N0030 G1 X10  Y50  F900
N1000 G08 P0                ; Vorsteuerung ausschalten
```

Wird G08 ohne ”P” programmiert, wird der Alarm 12470 ausgegeben.

Um mit G08 P1 weitere Funktionen wie SOFT, BRISK etc. flexibel aktivieren zu können, wird mit G08 P.. der Zyklus CYCLE308.spf aufgerufen.

G08 P1 muß allein im Satz stehen.

2.2.23 Kompressor im ISO-Dialekt-Mode

Die Befehle COMPON, COMPCURV, COMPCAD sind Befehle der Siemenssprache und aktivieren eine Kompressorfunktion, die mehreren Linearsätze zu einem Bearbeitungsabschnitt zusammenfasst.
Wird diese Funktion im Siemensmode aktiviert, können jetzt auch Linearsätze im ISO-Dialekt-Mode mit dieser Funktion komprimiert werden.
Die Sätze dürfen maximal aus folgenden Befehlen bestehen:

- Satznummer
- G01, modal oder im Satz
- Achszuweisungen
- Vorschub
- Kommentare

Enthält ein Satz andere Kommandos (z.B. Hilfsfunktionen, andere G Codes usw.), wird nicht komprimiert.
Wertzuweisungen mit $x für G, Achsen und Vorschub sind möglich, ebenso die Funktion Skip.

Beispiel: Diese Sätze werden komprimiert

```
N5   G290
N10  COMPON
N15  G291
N20  G01 X100. Y100. F1000
N25  X100 Y100 F$3
N30  X$3 /1 Y100
N35  X100 (Achse 1)
```

Diese Sätze werden **nicht** komprimiert

```
N5   G290
N10  COMPON
N20  G291
N25  G01 X100 G17          ;G17
N30  X100 M22              ;Hilfsfunktion im Satz.
N35 X100 S200              ;Spindeldrehzahl im Satz
```

2.2.24 Automatischer Eckenoverride G62

An Innenecken mit aktiver Werkzeugradiuskorrektur ist es oft sinnvoll, den Vorschub zu verringern.

G62 wirkt nur an Innenecken mit aktiver Werkzeugradiuskorrektur und aktivem Bahnsteuerbetrieb. Es werden nur Ecken berücksichtigt, deren Innenwinkel kleiner ist als MD 42526: $SC_CORNER_SLOWDOWN_CRIT. Der Innenwinkel ist bestimmt aus dem Knick in der Kontur.

Der Vorschub wird um den Faktor 42524: $SC_CORNER_SLOWDOWN_OVR abgesenkt:

gefahrener Vorschub =
F * $SC_CORNER_SLOWDOWN_OVR * Vorschuboverride.

Der Vorschuboverride setzt sich zusammen aus der Vorschubkorrektur von der Maschinensteuertafel multipliziert dem Override aus Synchronaktionen.

Die Vorschubabsenkung wird begonnen mit einem Abstand von 42520: $SC_CORNER_SLOWDOWN_START vor der Ecke. Sie endet mit dem Abstand 42522: $SC_CORNER_SLOWDOWN_END nach der Ecke (siehe Bild LEERER MERKER). An gekrümmten Konturen wird ein entsprechender Bahnweg verwendet.

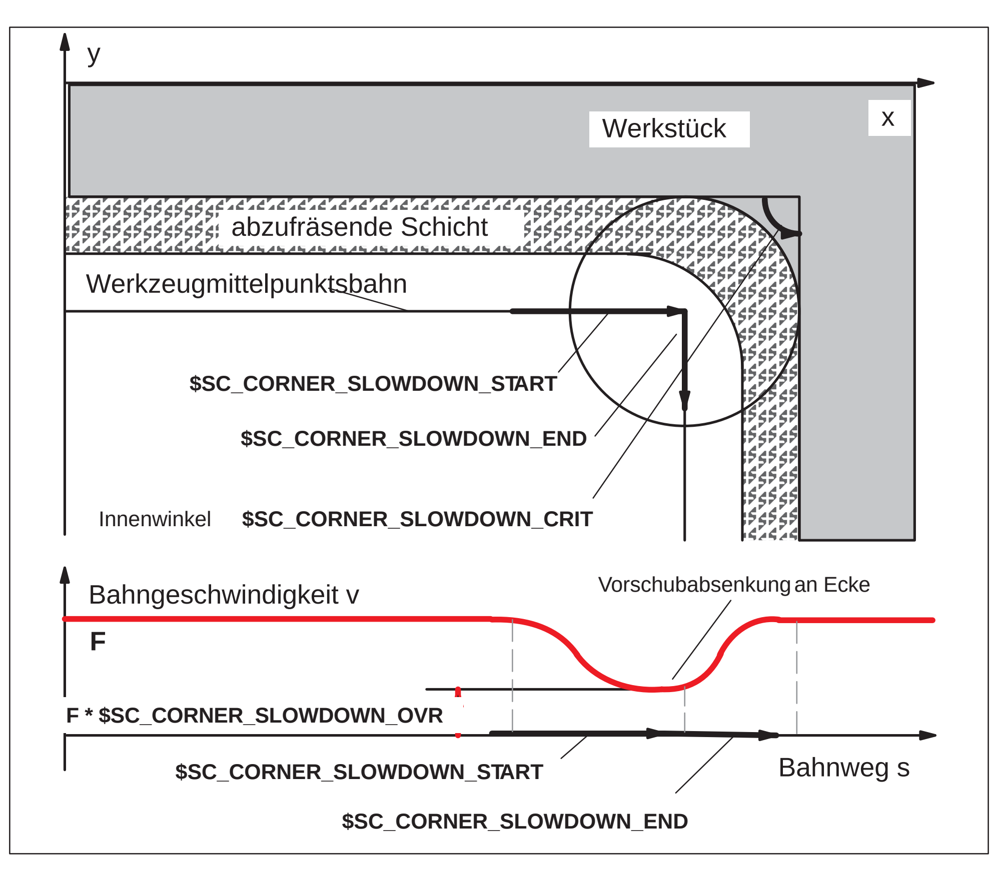


Bild 2-8 Parametrierung der vorschubreduzierung G62 am Beispiel einer 90°-Ecke

Parametrierung

Der Overridewert wird über folgende Settingdaten eingestellt:

42520: $SC_CORNER_SLOWDOWN_START
42522: $SC_CORNER_SLOWDOWN_END
42524: $SC_CORNER_SLOWDOWN_OVR
42526: $SC_CORNER_SLOWDOWN_CRIT

Die Settingdaten werden mit dem Wert 0 vorbelegt.

- Wenn $SC_CORNER_SLOWDOWN_CRIT == 0, wirkt die Eckenverzögerung dann nur an Reversierpunkten.
- Sind $SC_CORNER_SLOWDOWN_START und $SC_CORNER_SLOW-DOWN_END gleich 0, so wird die Vorschubreduzierung mit der zulässigen Dynamik angefahren.
- Ist $SC_CORNER_SLOWDOWN_OVR == 0, so wird ein kurzzeitiger Stopp eingefügt.
- $SC_CORNER_SLOWDOWN_CRIT bezieht sich bei G62 auf die Geometrieachsen. Es definiert den maximalen Innenwinkel in der aktuellen Bearbeitungsebene, bis zu dem die Eckenverzögerung angewendet wird. – G62 ist nicht bei Eilgang wirksam.

Aktivierung

Die Funktion wird aktiviert über G62 bzw. G621. Der G–Code wird entweder über den entsprechenden Teileprogrammbefehl oder über MD 20150: $MC_GCODE_RESET_VALUES[56] aktiv.

Beispiele

```
$TC_DP1[1,1]=120
$TC_DP3[1,1]=0.            ;Laengenkorrektuvektor
$TC_DP4[1,1]=0.
$TC_DP5[1,1]=0.

N1000 G0 X0 Y0 Z0 F5000 G64 SOFT
N1010 STOPRE
N1020 $SC_CORNER_SLOWDOWN_START = 5.
N1030 $SC_CORNER_SLOWDOWN_END = 8.
N1040 $SC_CORNER_SLOWDOWN_OVR = 20.
N1050 $SC_CORNER_SLOWDOWN_CRIT = 100.

N2010 G1 X00 Y30 G90 T1 D1 G64
N2020 G1 X40 Y0  G62 G41   ;Innenecke  zu N2030,
                           ;aber WRK noch in Anwahl
N2030 G1 X80 Y30           ;Innenecke zu N2040 127 Grad
N2040 G1 Y70               ;Innenecke zu N2050 53 Grad
N2050 G1 X40 Y40           ;Aussenecke zu N2060
N2060 G1 X20 Y70           ;Innenecke zu N2070 97 Gra
Y60           ;Innenecke zu N2080 90 Grad
N2080 G1 X20 Y20           ;Aussenecke zu N2090, irre-
                           levant, da WRK Abwahl
N2090 G1 X00 Y00 G40 FENDNORM

M30
```

2.3 Unterprogramm– und Makrotechnik

2.3.1 Unterprogrammtechnik: M98

Unterprogrammaufruf

Bei ISO-Dialekt werden Unterprogrammaufrufe mit M98 programmiert. Programmsyntax siehe Bild 2-9.

```
M98 P xxxxyyyy
       │   └── Programmnummer (max. 4 Stellen)
       └────── Anzahl Wiederholungen (max. 4 Stellen)
```

Bild 2-9 Beschreibung der zulässigen Parameter

Mit der Programmsyntax M98 Pxxxxyyyy wird ein Unterprogramm mit der Nummer yyyy aufgerufen und xxxxmal wiederholt. Wird xxxx nicht programmiert, wird das Unterprogramm nur einmal ausgeführt. Der Unterprogrammname besteht immer aus 4 Ziffern oder wird durch das Zufügen der 0 auf 4 Ziffern erweitert.
Wird z.B. M98 P21 programmiert, wird im Teileprogrammspeicher nach dem Programmnamen 0021.spf gesucht und das Unterprogramm 1mal ausgeführt. Soll das Unterprogramm 3mal ausgeführt werden, ist M98 P30021 zu programmieren.

ab Softwarestand 6

Bisher wird die Programmdurchlaufzahl (Anzahl der Wiederholungen) im ISO–Dialekt–M/T zusammen mit der Unterprogrammnummer unter der Adresse “P” programmiert.

Alternativ zu dieser Programmiersyntax kann die Unterprogrammdurchlaufzahl jetzt auch unter der Adresse “L” programmiert weden. Die Nummer des Unterprogrammes wird weiterhin mit Pxxxx programmiert. Wird die Durchlaufzahl mit beiden Adressen programmiert, gilt die Durchlaufzahl, die mit der Adresse “L” programmiert ist. Die Unterprogrammdurchlaufzahl kann zwischen 1 und 9999 liegen.

Beispiel:

```
N20 M98 P20123       ;das Unterprogramm 123.spf wird zweimal ausgeführt
N40 M98 P55 L4       ;das Unterprogramm 55.spf wurde viermal ausgeführt
N60 M98 P30077 L2    ;das Unterprogramm 77.spf wird zweimal ausgeführt.
                     ; Die unter Adresse “P” programmierte Durchlaufzahl =3
                     ; wird ignoriert.
```

Unterprogramm beenden

Die Beendigung des Unterprogrammes erfolgt mit M99.
Ist M99 Pxxxx programmiert, wird beim Rücksprung in das Hauptprogramm zu der Satznummer Nxxxx verzweigt. Die Satznummer muß immer mit “N” beginnen. Die Satznummer wird zunächst vorwärts (vom Unterprogrammaufruf in Richtung Programmende) gesucht. Wird keine passende Satznummer gefunden, wird das Teileprogramm anschließend rückwärts (in Richtung Teileprogrammanfang) durchsucht.

Steht M99 ohne Satznummer (Pxxxx) in einem Unterprogrammprogramm wird das Unterprogramm beendet und im Hauptprogramm auf den Satz nach dem Unterprogrammaufruf gesprungen.

Steht M99 ohne Satznummer (Pxxxx) in einem Hauptprogramm wird auf den Programmanfang des Hauptprogramms gesprungen und das Hauptprogramm erneut abgearbeitet.

Diese M-Befehle werden nicht an die PLC ausgegeben.

Unterprogrammrücksprung mit “RET”

Gilt nur bei ISO-Dialekt-T.

In den Siemenshüllzyklen für das Abspanen ist es nach dem Schruppen notwendig (wie bei ISO-Dialekt), mit der Programmbearbeitung im Hauptprogramm nach der Konturbeschreibung fortzufahren. Dazu muß der Hüllzyklus einen Unterprogrammrücksprung auf den Satz nach Ende der Konturbeschreibung ausführen. Um bei den Abspanzyklen nach dem Unterprogrammaufruf (mit G71-G73) die Sätze mit der Konturbeschreibung zu überspringen, wird der RET-Befehl um zwei optionale Parameter erweitert.

Mit dem Befehl RET (STRING: <Satznummer/Label>) wird die Programmbearbeitung im aufrufenden Programm (Hauptprogramm) mit dem Satz mit <Satznummer/Label> fortgesetzt.

Soll die Programmbearbeitung mit dem nächsten Satz nach <Satznummer/Label> fortgesetzt werden, muß der 2. Parameter im RET-Befehl > 0 programmiert werden RET (<Satznummer/Label>, 1). Wird für den 2. Parameter ein Wert > 1 programmiert, erfolgt der Unterprogrammrücksprung auch auf den Satz nach dem Satz mit <Satznummer/Label>.

Bei den Zyklen G70-G73 ist die zu bearbeitende Kontur im Hauptprogramm abgelegt. Um am Ende von G70 (Schlichtschnitt über die Kontur mit dem Abspanzyklus) im Hauptprogramm nach der Konturbeschreibung aufzusetzen, wird der erweiterte RET-Befehl benötigt. Um mit dem Ende des Hüllzyklus für G70 auf den nächsten NC-Satz nach der Konturbeschreibung zu springen, muß er mit folgender Returnsyntax beendet werden:

RET (“N” << $C_Q, 1)

Suchrichtung:

Die Suchrichtung für <Satznummer/Label> erfolgt immer zuerst in Vorwärtsrichtung (Richtung Programmende) und dann in Rückwärtsrichtung (Richtung Programmanfang).

Beispiel

```
N10 X10. Y20.
N20  G71 P30  Q60 U1 W1 F1000 S1500
   N10 ...                        ;Hüllzyklus für Abspanzyklus
   N20 DEF STRING[6]BACK
   N30 ...
   N90
   N100 RET (”N”<<$C_Q, 1)        ;Rückspr. auf Satz nach
                                  ;Konturbeschr. -> N70
N30 X50. Z20.
N40 X60.
N50 Z55.
N60 X100. Z70.
N70 G70 P30 Q60
N80 G0 X150. Z200.
N90 M30
```

Hinweis

M30 im Siemens-Mode: wird im Unterprogramm als Programm-Rücksprung interpretiert.

M30 im ISO-Dialekt-Mode: wird auch im Unterprogramm als Teileprogramm-ende interpretiert.

2.3.2 Siemenssprachbefehle im ISO-Dialekt-Mode

Für Shopmill werden bestimmte Siemenssprachbefehle auch im ISO-Dialekt-Mode benötigt, die dann im ISO-Dialekt-Mode ausgeführt werden. Zu diesen Sprachbefehlen zählen Unterprogrammaufrufe mit und ohne Parameterübergabe (nicht Aufrufe mit Lxx, da die Adresse L bei ISO-Dialekt bereits eine andere Bedeutung hat), Programmteilwiederholung und Kontrollstrukturen. Alle anderen Siemenssprachbefehle werden im ISO-Dialekt-Mode mit einem Alarm abgelehnt.

Im aktiven ISO-Dialekt-Mode dürfen folgende Siemenssprachbefehle programmiert werden:

REPEAT:

REPEAT	Satznummer> [<Satznummer>] [P..]
REPEAT	UNTIL
REPEATB	<Satznummer> [P..]

Als Start- und Endekennung der Programmteilwiederholung sind nur Satznummern, keine Labels erlaubt.

IF – ELSE – ENDIF
FOR – ENDFOR
WHILE – ENDWHILE
IF<Bedingung> **– GOTO F**<Bedingung>
CASE

Modale und nichtmodale Unterprogrammaufrufe

```
N100  CALL ”WELLE” oder
N100  MCALL WELLE oder
N100  WELLE
```

Modale und nichtmodale Unterprogrammaufrufe mit Parameterübergabe

```
N100  MCALL WELLE (”ABC”; 33.5) oder
N100  WELLE (”ABC”; 33.5) Unterprogrammaufrufe mit Pfadangabe
N100  CALL”/_N_SPF_DIR/WELLE oder
N100  MCALL/_N_SPF_DIR/ WELLE oder
N100  PCALL/_N_SPF_DIR/WELLE
```

2.3.3 Erweiterung des Unterprogrammaufrufs für Konturaufbereitung mit CONTPRON

Bei ISO-Dialekt steht die Konturbeschreibung für die Abspanzyklen G70-G73 nicht gesondert in einem Unterprogramm (wie bei SINUMERIK), sondern ist im Teileprogramm (Hauptprogramm) abgelegt. Beim Aufruf der Zyklen wird der Bereich der Konturbeschreibung mit einer Anfangs- und Endesatznummer definiert. Die Zyklen erhalten diese Satznummer als Übergabeparameter. Für die Siemens-Anpaßzyklen wird der indirekte Unterprogrammaufruf erweitert.

Bisher wird ein Unterprogrammaufruf indirekt mit CALL <Programmname> aufgerufen.

Für den Zugriff auf die Konturbeschreibung im Hauptprogramm wird der indirekte Unterprogrammaufruf wie folgt erweitert:

CALL [<Programmname>] BLOCK <Startlabel> TO <Endelabel>

Wird kein Programmname oder ein Leerstring als Programmname angegeben, also CALL BLOCK <Startlabel> TO <Endelabel>, wird der Programmteil (Start-/Endelabel) im aktuell angewählten Programm gesucht. Auch bei MDA, ASUP usw. werden die Labels im angewählten Programm gesucht (z.B. im Fall von MDA werden die Labels nicht im MDA-Puffer sondern in dem Programm mit dem angewählten Programmnamen gesucht). Wird diese Syntax direkt in ein Hauptprogramm geschrieben, wirkt sie wie eine Programmteilwiederholung mit REPEAT <Startlabel> <Endelabel> mit einem Schleifendurchlauf, d.h. Start- und Endelabel werden in dem selben Programm gesucht, in dem auch der CALL BLOCK ... - Befehl programmiert ist.

Wird ein Programmname angegeben, also CALL <progName> BLOCK <Startlabel> TO <Endelabel>, wird der Programmteil (Start-/Endelabel) in dem Unterprogramm “progName” gesucht.

Beispiel

```
Nxx G71 Pxx Q1110 U.. W..   ;ISO-Dialekt-G-Funktion ruft Hüllzyklus
                            ; CYCLE395.spf auf
              ;_N_CYCLE395_SPF
              N10 .......
              .......
              Nxxx CYCLE95(....., "N"<<$C_P, "N"<<$C_Q)
                            ;Abspanzyklus mit zusätzlichen Parameter
                            ; für Start- und Endelabel
              PROC CYCLE95(....., STRING[20] startlab, STRING[20]
                            endelab)
              N10 ........
              .........
              Nxxx CONTPRON(...)

              N.... CALL "" BLOCK startlab TO endelab
                            ;  Konturbeschreibung lesen oder
              N.... CALL BLOCK startlab TO endelab
                            ;Aufruf des Konturprogramms
              EXECUTE(...)
              ........
              Nxx  M17
Nxxx .....
Nxxx RET ("N"<<$C_Q, 1)     ;Rücksprung auf den nächsten Satz nach
                            ; der Konturbeschreibung
N1120 ....

Nxxx M30
```

Hinweis

CONTPRON- und EXECUTE-Aufruf selbst müssen nicht verändert werden.

Anfangssatznummer suchen

Die Suche nach der Anfangssatznummer (Startlabel) der Konturbeschreibung erfolgt immer zuerst in Richtung Programmende (vorwärts) und dann in Richtung Programmanfang (rückwärts).

Wird die gleiche Satznummer mehrmals programmiert, wird immer die nächste Satznummer (Label) nach dem aktuellen Satz im Programm in dem die Konturbeschreibung steht als Beginn der Konturbeschreibung erkannt (siehe Beispiel). Der aktuelle Satz ist in der Regel der Satz, in dem der Abspanzyklus (Hüllzyklus) im Hauptprogramm aufgerufen wurde.

Beispiel

Im Abspanzyklus CYCLE395 soll die Konturbeschreibung, die im Hauptprogramm zwischen den Sätzen `N10 - N30` steht, verwendet werden (mit `CALL BLOCK N10 TO N30` in CYCLE395). `N40` ist die aktuelle Programmzeile im Hauptprogramm.

Der Konturbeschreibungsblock ist im Beispiel **fett** gedruckt.

```
N10 X10. Y20.N20 X30.
N30 Y10.
N40 G71 P10 Q30...        ;Hüllzyklus für Abspanzyklus aufrufen
...                       ;(Im  Abspanzyklus wird
...                       ;"CALL BLOCK N10 TO N30" programmiert)
...                       ;als Konturbeschreibung werden die fettgedruckten
                          ; Zeilen gefunden

N50 G90 G54
N60 F1000 G94
N10 X50. Y10.
N20 X33. Y11.
N30 X10.
N50 ....
N..  .....
N800 G71 P10 Q30          ;Hüllzyklus für Abspanzyklus aufrufen
...                       ;(Im  Abspanzyklus wird "CALL BLOCK N10 TO N30"
                          ; programmiert)
...                       ;als Konnturbeschreibung werden die
...                       ;kursiv ;gedruckten Zeilen gefunden

N999 ....
N10 X15.
N20 Y25.
N30 X33.
N1010 ....
N1020 .....
```

2.3.4 Makrobefehle mit G65, G66 und G67

ISO-Dialekt

Im ISO-Dialekt-Mode werden mit G65 Pxx, G66 Pxx im Teileprogramm Makros aufgerufen. Als Makro bezeichnet man die Zusammenfassung von mehreren Teileprogrammsätzen, die mit M17 abgeschlossen werden.

Beim Aufruf des Unterprogrammes wird vom ISO Mode in den Siemens Mode geschaltet.

Die An- und Abwahl erfolgt mit den folgenden Befehlen:

- G65 Makroaufruf, satzweise wirksam
- G66 Makroaufruf, modal wirksam
- G67 modalen Makroaufruf abwählen

Siemens

Mit den G-Befehlen G65 Pxx und G66 Pxx wird das Makro xx gestartet. Mit G65 wird das Unterprogramm Pxx einmal aufgerufen, mit G66 wirkt der Unterprogrammaufruf Pxx modal und wird in jedem Satz mit Achsbewegungen ausgeführt (wie MCALL xx). Mit G67 wird der modale Unterprogrammaufruf wieder abgewählt (entsprechend G80 bei Zyklenaufrufen).

In einem Teileprogrammsatz mit G65 oder G66 wird die Adresse Pxx als Programmnummer des Unterprogramms interpretiert, in dem die Makrofunktionalität programmiert ist. Mit der Adresse Lxx kann die Durchlaufzahl der Makros definiert werden. Ist im Aufrufsatz keine Durchlaufzahl programmiert, wird das Makro einmal ausgeführt. Alle weiteren Adressen in diesem Teileprogrammsatz werden wie bei ISO-Dialekt-”Makro B” als Übergabeparameter interpretiert und deren programmierte Werte in den Systemvariablen $C_A - $C_Z gespeichert. In den Unterprogrammen können diese Systemvariablen gelesen und für die Makrofunktionalität ausgewertet werden. Werden in einem Makro (Unterprogramm) weitere Makros mit Parameterübergabe aufgerufen, müssen die Übergabeparameter im Unterprogramm vor dem neuen Makroaufruf in internen Variablen gespeichert werden.

Damit interne Variablendefinitionen möglich sind, wird wie bei den Bearbeitungszyklen der Sprachmodus implizit auf Siemensmode umgestellt. Wird im Unterprogramm ein weiterer Makroaufruf programmiert, muß deshalb vorher wieder ISO-Dialekt-Mode angewählt werden.

Systemvariablen für die Adressen I, J, K

Da die Adressen I, J, und K bis zu zehnmal in einem Satz mit Makroaufruf programmiert werden können, muß auf die Systemvariablen für diese Adressen mit einem Arrayindex zugegriffen werden. Die Syntax für diese drei Systemvariablen ist dann $C_I[..], $C_J[..], $C_K[..]. Die Werte stehen in der programmierten Reihenfolge im Array. Die Anzahl der im Satz programmierten Adressen I, J, K steht in den Variablen $C_I_NUM, $C_J_NUM, $C_K_NUM.

Die Übergabeparameter I, J, K für Makroaufrufe werden jeweils als zusammengehörender Block behandelt, auch wenn einzelne Adressen nicht programmiert werden. Wenn ein Parameter erneut programmiert wird, oder ein nachfolgender Parameter bezogen auf die Reihenfolge I, J, K programmiert wurde, so gehört er zum nächsten Block.

Um im ISO Mode die Programmierreihenfolge zu erkennen, werden die Systemvariablen $C_I_ORDER, $C_J_ORDER, $C_K_ORDER gesetzt. Diese sind identische Arrays zu $C_I, $C_K und enthalten die zugehörige Nummer zum Parameter.

Beispiel:

```
N5 I10 J10 K30 J22 K55 I44 K33
    set1        set2     set3
$C_I[0]=10
$C_I[1]=44
$C_I_ORDER[0]=1
$C_I_ORDER[1]=3

$C_J[0]=10
$C_J[1]=22
$C_J_ORDER[0]=1
$C_J_ORDER[1]=2

$C_K[0]=30
$C_K[1]=55
$C_K[2]=33
$C_K_ORDER[0]=1
$C_K_ORDER[1]=2
$C_K_ORDER[2]=3
```

Zyklenparameter $C_x_PROG

Bei ISO–Dialekt–0 Mode können die programmierten Werte abhängig von der Programmierweise (Integer- oder Real-Wert), unterschiedlich bewertet werden. Die unterschiedliche Bewertung wird über ein Maschinendatum aktiviert.

Ist das MD gesetzt, verhält sich die Steuerung wie im folgenden Beispie:

X100. ;X–Achse wird 100 mm verfahren (100. mit Punkt => Real–Wert
Y200 ;Y–Achse wird 0,2 mm verfahren (200 ohne Punkt => Integer–Wert

Werden die im Satz programmierten Adressen als Übergabeparameter für Zyklen verwendet, stehen die programmmierten Werte immer als Realwerte in den $C_x-Variablen. Bei ganzzahligen Werten ist in den Zyklen kein Rückschluß auf die Programmierweise (Real/Integer) mehr möglich und damit auch keine Bewertung des programmierten Wertes mit dem richtigen Umrechnungsfaktor.

Für die Information, ob REAL oder INTEGER programmiert wurde, gibt es die Systemvariable $C_TYP_PROG. $C_TYP_PROG ist genauso aufgebaut wie $C_ALL_PROG und $C_INC_PROG. Für jede Adresse (A–Z) gibt es ein Bit. Ist der Wert als INTEGER programmiert, wird das Bit 0 gesetzt, bei REAL auf 1. Ist der Wert über eine Variable $<Nummer> programmiert, wird Bit 2 = 1 gesetzt.

Beispiel:

M98 A100. X100 –> $C_TYP_PROG == 1.
Es sitzt nur das Bit 0, da nur A als REAL programmiert wurde.

M98 A100. C20. X100 –> $C_TYP_PROG == 5.
Es sitzt das Bit 1 und 3 (A und C).

Einschränkungen:

In jedem Satz können maximal zehn I, J, K Parameter programmiert werden. In der Variablen $C_TYP_PROG ist für I, J, K nur jeweils ein Bit vorgesehen. Daher ist in $C_TYP_PROG für I, J und K das Bit 2 immer auf 0 gesetzt. Es läßt sich also nicht ableiten, ob I, J oder K als REAL oder INTEGER programmiert sind.

Die Parameter P, L, O, N können nur als Integer programmiert werden. Ein Realwert führt zu einem NC–Alarm. Daher ist das Bit in $C_TYP_PROG immer 0.

Modale Makroaufrufe

Bei modalen Makroaufrufen werden die programmierten Adressen nur im Aufrufsatz (Satz mit G66) in die Systemvariablen geschrieben. Danach wird das Makro in jedem Satz mit Achsbewegung ausgeführt, bis eine Abwahl mit G67 oder ein neuer Makroaufruf mit G66 programmiert wird. Bei modalen Makros werden im Aufrufsatz (= Satz mit G66) nur die Makroparameter versorgt. Die Ausführung des Makros erfolgt zum erstenmal im nächsten Satz mit Achsbewegung. (Ablauf wie bei MCALL xx im Siemensmode.)

Beispiel Makroaufruf:

```
_N_M10_MPF:

N10 M3 S1000 F1000
```

```
N20 X100. Y50. Z33.
N30 G65 P10 F55 X150. Y100. S2000
N40 X50.
N50 ....
N200 M30
```

Beispiel für ein Unterprogramm als Makro im Siemens-Mode:

```
_N_10_SPF:
N10 DEF REAL X_ACHSE, Y_ACHSE, DREHZAHL, VORSCHUB
N15 X_ACHSE = $C_X Y_ACHSE = $C_Y  DREHZAHL = $C_S VORSCHUB = $C_F
N20 G01 F=VORSCHUB  G95 S=DREHZAHL
...
M17
```

2.3.5 Modeumschaltung bei Makroaufrufen mit G65/G66

Bisher wurde bei Makroaufrufen mit G65/G66 automatisch in den Siemensmode gewechselt.
Nun kann der Anwender entscheiden, ob am Makroanfang in den Siemensmode gewechselt werden soll. Steht in der ersten Programmzeile des Makroprogramms die Anweisung PROC<Programmname>, wird in den Siemensmode gewechselt. Fehlt diese Anweisung, bleibt der ISO–Mode auch im Makroprogramm aktiv.

Damit kann der Anwender entscheiden, ob er lokale Variablen (mit DEF...) anlegen will, um Übergabevariablen zu speichern. Dazu muß er mit der PROC–Anweisung in den Siemensmode schalten. Er kann aber auch entscheiden, ob sein Makroprogramm (z.B. ein bestehendes ISO–Dialekt–M/T Makro) im Iso Mode abgearbeitet werden kann.

Beispiel für den Makroaufruf:

```
_N_M10_MPF:
N10 M3 S1000 F1000
N20 X100. Y50. Z33.
N30 G65 P10 F55 X150. Y100. S2000
N40 X50.
N50....
N200 M30
```

Beispiel für ein Unterprogramm als Makro im Siemensmode:

```
_N_0010_SPF:
PROC 0010    ;Umschaltung in den Siemensmode
N10 DEF REAL X_ACHSE, Y_ACHSE, DREHZAHL, VORSCHUB
N15 X_ACHSE=$C_X Y_ACHSE=$C_Y DREHZAHL=$C_S VORSCHUB=$C_F
N20 G01 F=VORSCHUB G95 S=DREHZAHL
....
N80 M17
```

Beispiel für ein Unterprogramm als Makro im ISO Mode:

```
_N_0010_SPF:
G290          ;Umschaltung in den Siemensmode

              ;falls Übergabevariable gelesen werden müssen
N15 X_ACHSE=$C_X Y_ACHSE=$C_Y DREHZAHL=$C_S
N20 G01 F=$C_F G95 S=$C_S
N10 G1 X=$C_X Y=$C_Y
G291          ;Umschalten in den ISO Mode
N15 M3 G54 T1
N20
....
N80 M99
```

2.3.6 Makroaufruf über M-Funktion

Über M-Nummern kann analog zu G65 (siehe 2.3.5) ein Makro aufgerufen werden.

Die Projektierung von 10 M-Funktionsersetzung erfolgt über die Maschinendaten
10814: $MN_EXTERN_M_NO_MAC_CYCLE und
10815: $MN_EXTERN_M_NO_MAC_CYCLE_NAME.

Die Parameterübergabe erfolgt identisch zu G65. Wiederholungen können mit der Adresse L programmiert werden.

Einschränkungen

Pro Teileprogrammzeile kann nur eine M-Funktionsersetzung (bzw. nur ein Unterprogrammaufruf) ausgeführt werden. Konflikte mit anderen Unterprogrammaufrufen werden mit Alarm 12722 gemeldet. In dem ersetzten Unterprogramm erfolgt keine weitere M-Funktionsersetzung.

Es gelten sonst die gleichen Einschränkungen wie bei G65

Konflikte mit vordefinierten und anderen definierten M–Nummern werden mit Alarm abgelehnt.

Projektierbeispiele

Aufruf des Unterprogramms M101_MAKRO durch die M-Funktion M101

$MN_EXTERN_M_NO_MAC_CYCLE[0] = 101
$MN_EXTERN_M_NO_MAC_CYCLE_NAME[0] = ”M101_MAKRO”

Aufruf des Unterprogramms M6_MAKRO durch die M-Funktion M6.

$MN_EXTERN_M_NO_MAC_CYCLE[1] = 6
$MN_EXTERN_M_NO_MAC_CYCLE_NAME[1] = ”M6_MAKRO”

Programmierbeispiel für Werkzeugwechsel mit M-Funktion:

```
PROC MAIN
...
N10          M6 X10 V20
...
N90          M30

PROC M6_MAKRO

...
N0010        R10 = R10 + 11.11
N0020        IF $C_X_PROG == 1 GOTOF N40
display($C_X_PROG)
N0030        SETAL(61000)  ;programmierte Variable nicht richtig
                           ;übergeben
N0040        IF $C_V == 20 GTOF N60
display($C_V)
N0050        SETAL(61001)
N0060        M17
```

2.3.7 Makroaufruf über G–Funktion

Über eine G-Nummer kann analoh zu G65 (siehe 2.3.5) ein Makro aufgerufen werden.

Es sind 50 G–Funktionsersetzungen über die Maschinendaten
10816: $MN_EXTERN_G_NO_MAC_CYCLE und
10817: $MN_EXTERN_G_NO_MAC_CYCLE_NAME
projektierbar.

Die im Satz programmierten Parammeter werden in den $C_–Variablen abgelegt. Mit der Adresse L wird die Anzahl der Makrowiederholungen programmiert. In der Variablen $C_G wird die Nummer des programmierten G_Makros abgelebt. Alle weiteren im Satz programmierten G–Funktionen werden wie normale G–Funktionen behandelt. Die Programmierreihenfolge der Adressen und G–Funktionen im Satz ist beliebig und hat keine Auswirkung auf die Funktionalität.

Alle ISO–G–Codes, auch G–Codes mit Dezimalpunkt (= Realwert) können durch einen Makroaufruf ersetzt werden.

G–Funktionen , die durch ein Makro ersetzt werden, sind in der Steuerung noch vorhanden und können über
10882: $MN_NC_USER_EXTERN_GCODES_TAB[ ] umdefiniert werden.

Einschränkungen

Pro Teileprogrammzeile kann nur eine G/M-Funktionsersetzung (bzw. nur ein Unterprogrammaufruf) ausgeführt werden. Konflikte mit anderen Unterprogrammaufrufen, z.B. wenn ein modaler Unterprogrammaufruf aktiv ist, werden mit Alarm 12722 gemeldet.

Ist ein G–Makro aktiv, wird kein weiteres G/M–Makro oder M–Unterprogramm aufgerufen. M–Makros/–Unterprogramme werden dann als M–Funktionen ausgeführt, G–Makros als G–Funktion, wenn eine entsprechende G–Funktion existiert. Andernfalls wird der Alarm 12470 ausgegeben.

Es gelten sonst die gleichen Einschränkungen wie bei G65

Projektierbeispiele

Aufruf des Unterprogramms G21_MAKRO durch die G-Funktion G21

$MN_EXTERN_G_NO_MAC_CYCLE[0] = 21
$MN_EXTERN_G_NO_MAC_CYCLE_NAME[0] = ”G21_MAKRO”

$MN_EXTERN_G_NO_MAC_CYCLE[1] = 123
$MN_EXTERN_G_NO_MAC_CYCLE_NAME[1] = ”G123_MAKRO”
$MN_EXTERN_G_NO_MAC_CYCLE[2] = 421
$MN_EXTERN_G_NO_MAC_CYCLE_NAME[2] = ”G123_MAKRO”

Programmierbeispiel:

```
PROC MAIN
...
N0090 G291                        ;ISO–Mode
N0100 G1 G21 X10 V20 F1000 G90    ;Aufruf von G21_MAKRO.spf,
                                  ;G1 und G90 werden vor dem
                                  ;Aufruf von G21_MAKRO.spf
                                  ;aktiviert
...
N0500 G90 X20 Y30 G123 G1 G54     ;Aufruf von G123_MAKRO.spf,
                                  ;G1, G54 und G90 werden vor
                                  :dem Aufruf von G123_MA-
                                  ;KRO.spf aktiviert
...
N0800 G90 X20 Y30 G421 G1 G54     ;Aufruf von G421_MAKRO.spf,
                                  ;G1, G54 und G90 werden vor
                                  :dem Aufruf von G123_MA-
                                  ;KRO.spf aktiviert
...
N0900 M30

PROC G21_MAKRO
...
N0010  R10 = R10 + 11.11
N0020  IF $C_X_PROG == 0
N0030  SETAL(61000)              ;programmierte Variable nicht rich-
                                 ;tig übergeben
N0040  ENDIF
N0050  IF $C_V_PROG == 0
N0060  SETAL(61001)
N0070  ENDIF
N0080  IF $C_F_PROG == 0
N0090  SETAL(61002
N0100  ENDIF
N0110 G90 X=$C_X V=$C_V
N0120 G291
N0130 G21 M6 X100                ;G21->Maßsystem metrisch aktivieren
                                  (kein Makroaufruf)
N0140 G290
...
N0150 M17

PROC G123_MAKRO
...
N0010  R10 = R10 + 11.11
N0020  IF $C_G == 421 GOTOF label_G421
                       ;Makrofunktionalität für G123
N0040 G91 X=$C_X Y=$C_Y F500
...
```

```
...
N1990  GOTOF label_ende
N2000 label_G421:      ;Makrofunktionalität für G421jN2010 G90 X=$C_X
Y=$C_Y F100
N202
...
...
N3000 G291
N2010 G123             ;Alarm 12470, da G123 keine G–Funktion ist und
                       ;ein Makroaufruf bei aktivem Makro nicht mög-
                       ;lich ist. Ausnahme: das Makro wurde als
                       ;Unterprogramm mit CALL G123_MAKRO aufgerufen.
N4000 label_ende: G290
N4010 M17
```

2.3.8 High-speed cycle cutting G05 P..

G05 P.. high–speed cycle cutting ist als Unterprogrammaufruf realisiert.

Programmierung G05 P.. L..

Pxxxxx	Unterprogrammnummer, max. 10 Stellen Beim Aufruf werden keine 0 aufgefuellt wie bei M98.
Lxxxx	Durchlaufzahl, wenn kein L programmiert ist wirkt L1.

Beispiel:
G05 P10123 L3 10123.mpf wird 3mal durchlaufen.

Mit diesem Aufruf wird ein beliebiges Unterprogramm aufgerufen. Dieses Unterprogramm kann ein vorkompaliertes Programm sein, muß aber nicht. Es können jedoch nur Siemens Teileprogramm vorübersetzt werden.

Für die ISO-Dialekt-Funktion G05 gibt es keine Entsprechung im Siemensmode. Mit dem CYCLE305 hat der Anwender die Möglichkeit, eine im Rahmen der Siemensfunktionalität, eigene Funktionalität zu programmieren.

In folgenden Fällen wird bei der Prgrammierung von G05 der Zyklus CYCLE305.spf aufgerufen:

- G05 ohne P im Satz wird ohne Alarm überlesen.
- G05.1 mit und ohne P wird ohne Alarm überlesen.
- G05 P0 oder P01 sind reserviert für high–speed Remote buffer B. Diese Funktion wird nicht unterstützt.

Alle im Satz programmierten Adressen werden in den genannten Fällen in die Zyklenparameter $C_xx geschrieben. Beim Aufruf von CYCLE305 wird keine automatische Modeumschaltung von ISO nach Siemens durchgeführt. Soll CYCLE305.spf im Siemensmode bearbeitet werden, muß in der ersten Programmzeile eine PROC-Anweisung stehen wie bei Makroaufrufen mit G65/G66.

Alle programmierten funktionen im Satz werden, wie bisher bei der Programmierung von G05, ausgeführt, d.h. programmierte Achsen weden verfahren, Hilfsfunktionen ausgegeben usw. Die programmierten Adressen werden nur als zusätzliche Information in die Zyklenparameter geschrieben.

Ist G05 zusammen mit einem Unterprogrammaufruf (M98 P..) im selben Satz programmiert, wird der Alarm 12722 ausgegeben.

2.3.9 Umschaltmodi für DryRun und Ausblendebenen

Das Umschalten der Ausblendebenen (DB21 DBB2) stellt immer einen Eingriff in den Programmablauf dar, der bislang zu einem kurzfristigen Geschwindigkeitseinbruch auf der Bahn geführt hat. Gleiches gilt für das Umschalten des DryRun–Modus (DryRun = Probelaufvorschub DB21.DBB0.BIT6) von DryRunOff nach DryRunOn oder umgekehrt.

Mit einen neuen Umschaltmodus, der in seiner Funktion eingeschränkt ist, kann jetzt der Geschwindigkeitseinbruch umgangen werden.

Mit der Maschinendatenbelegung 10706: $MN_SLASH_MASK==2 wird beim Wechsel der Ausblendebenen (d.h. ein neuer Wert in der PLC–>NCK–Chan Nahtstelle DB21.DBB2) kein Geschwindigkeitseinbruch mehr notwendig.

Hinweis

Der NCK bearbeitet Sätze in zwei Stufen, der Vor– und Hauptbearbeitung, ab (auch Vorlauf und Hauptlauf). Das Ergebnis der Vorbearbeitung wandert in den Vorlaufspeicher. Die Hauptbearbeitung entnimmt dem Vorlaufspeicher den jeweils ältesten Satz und fährt seine Geometrie ab.

Achtung

Mit dem Maschinendatenbelegung $MN_SLASH_MASK==2 wird beim Wechsel der Ausblendebenen die Vorbearbeitung umgeschaltet! Alle Sätze, die sich im Vorlaufspeicher befinden, werden mit der alten Ausblendebene abgefahren. Der Anwender hat in der Regel keine Kontrolle über die Füllhöhe des Vorlaufspeichers. Der Anwender sieht damit folgenden Effekt: **“Irgendwann” nach dem Umschalten wird die neue Ausblendebene wirksam!**

Hinweis

Der Teileprogrammbefehl STOPRE leert den Vorlaufspeicher. Schaltet man vor dem STOPRE die Ausblendebene um, so sind alle Sätze nach dem STOPRE sicher umgeschaltet. Analog gilt das für ein implizites STOPRE.

Das Umschalten des DryRun–Modus hat die analogen Einschränkungen.

Mit der Maschinendatenbelegung 10704: $MN_DRYRUN_MASK==2 wird beim Wechsel des DryRun–Modus kein Geschwindigkeitseinbruch notwendig. Allerdings wird auch hier nur die Vorverarbeitung umgeschaltet, die zu den oben genannten Einschränkungen führt. Daraus ergibt sich analog: **Achtung "Irgendwann" nach dem Umschalten des DryRun–Modes wird dieser auch aktiv!**

2.3.10 Achtstellige Programmnummer

Mit dem Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK, Bit6=1 wird eine achstellige Programmnummernanwahl aktiviert. Diese Funktion hat Einfluss auf M98 (siehe Kapitel 2.3.1), G65/66 (siehe Kapitel 2.3.5) und M96 Kapitel 2.2.17).

y: Programmdurchlaufzahl
x: Programmnummer

Unterprogrammaufruf

$MC_EXTERN_FUNCTION_MASK, Bit6 = 0
M98 Pyyyyxxxx oder
M98 Pxxxx Lyyyy
Programmnummer max. 4stellig
Ergaenzung der Programmnummer immer auf 4 Stellen mit 0

Bsp.:	M98 P20012	ruft 0012.mpf 2 Durchläufe
	M98 P123 L2	ruft 0123.mpf 2 Durchläufe

$MC_EXTERN_FUNCTION_MASK, Bit6 = 1
M98 Pxxxxxxxx Lyyyy
Es erfolgt keine Ergänzung mit 0, auch wenn die Programmnummer weniger als 4 Stellen hat.
Die Programmierung von Durchlaufzahl und Programmnummer in P(Pyyyyxxxxx) ist nicht möglich,
die Durchlaufzahl muß immer mit L programmiert werden!

Bsp.: M98 P123 — ruft 123.mpf 1 Durchlauf
M98 P20012 — ruft 20012.mpf 1 Durchlauf,
Achtung: das ist nicht mehr kompatibel zu ISO–Dialekt–Original

M98 P12345 L2 — ruft 12345.mpf 2 Durchlaeufe

Modales und blockweises Makro G65/G66

$MC_EXTERN_FUNCTION_MASK, Bit6 = 0
G65 Pxxxx Lyyyy
Ergänzung der Programmnummer immer auf 4 Stellen mit 0. Programmnummer mit mehr als 4 Stellen führt zu Alarm.

$MC_EXTERN_FUNCTION_MASK, Bit6 = 1
G65 Pxxxx Lyyyy
Es erfolgt keine Ergänzung mit 0, auch wenn die Programmnummer weniger als 4 Stellen hat. Programmnummer mit mehr als 8 Stellen führt zu Alarm.

Interrupt M96

$MC_EXTERN_FUNCTION_MASK, Bit6 = 0
M96 Pxxxx
Ergänzung der Programmnummer immer auf 4 Stellen mit 0

$MC_EXTERN_FUNCTION_MASK, Bit6 = 1
M96 Pxxxx
Es erfolgt keine Ergänzung mit 0, auch wenn die Programmnummer weniger als 4 Stellen hat. Programmnummer mehr als 8 Stellen führt zu Alarm.

2.3.11 Systemvariable für Ebenenstack im ISO–Mode

Im Standardmode wird die aktuelle Programmebene in der Systemvariablen $P_STACK angezeigt. Jeder Unterprogrammaufruf und Rücksprung beeinflußt diese Variable. Im ISO–Mode gibt es jedoch Unterprogrammaufrufe, bei denen die aktuelle Ebene der Anwendervariablen **nicht** verändert wird. Um ebenspezifische Variablen über GUDs zu realisieren, ist es notwedig die aktuelle Programmebene im ISO–Mode zu kennen. Die Systemvariable $P_IPO_STACK liefert die aktuelle Programmebene im ISO–Dialekt–Mode.

In Tabelle 2-6 sind alle möglichen Unterprogramm- und Makroaufrufe im ISO–Mode und deren Einfluß auf die aktuelle Programmebene dargestellt.

Die Aufrufe im ISO–Mode werden auf Aufrufe im Standardmode abgebildet, so daß auch im ISO–Mode die Variable $P_STACK die selbe Information enthält wie bisher.
Die Anzahl der maximal möglichen Unterprogrammaufrufe bleibt unverändert.

Die Systemvariable $P_IPO_STACK wird immer inkrementiert, wenn ein Unterprogramm gestartet wird, das im ISO–Mode als Makroaufruf mit G65, G66, G–Code– oder M–Makro programmiert wird. Beim Rücksprung aus einem solchen ISO–Makro wird $P_IPO_STACK wieder dekrementiert. Ist kein ISO–Makro aktiv, ist $P_IPO_STACK = 0. $P_IPO_STACK liefert somit die Anzahl der gerade aktiven ISO–Makros.

Beim Aufruf eines mit M96 Pxx definierten Unterruptprogramms wird abhängig von dem MD $MC_EXTERN_FUNCTION_MASK Bit 11 die Variable $P_IPO_STACK ebenfalls inkremetiert.
$MC_EXTERN_FUNCTION_MASK
Bit 12 = 0, $P_IPO_STACK wird durch das Interruptprogramm nicht verändert.
Bit 12 = 1, $P_IPO_STACK wird durch das Interruptprogramm inkrementiert.

Zyklenaufrufe mit z.B. G81, G77 usw. oder Funktionen, die intern mit Zyklen realisiert sind, z.B. G05, G72.1 usw. und Unterprogrammaufrufe mit M98 Pxx beeinflussen $P_IPO_STACK nicht.

Beispiel: Unterprogrammaufrufe im ISO– und Standardmode.
M98 steht für Unterprogrammaufrufe ohne Erhöen der Ebene
G65 P steht für Makroaufruf mit Erhöhen der Ebene

Tabelle 2-6 Unterprogramm- und Makroaufrufe

$P_STACK	$P_IPO_STACK	Ebene 1	Ebene 2	Ebene 3
1	1	O111.mpf		
1	1	N5 M98 P2222		
2	1		O2222.mpf	
2	1		G65 P3333	
3	2			O3333.mpf
3	2			M99
2	1		M99	

Tabelle 2-6 Unterprogramm- und Makroaufrufe

$P_STACK	$P_IPO_STACK	Ebene 1	Ebene 2	Ebene 3
M98 schaltet die Ebenen nicht weiter. O1111.mpf und O2222.mpf arbeiten mit dem selben $P_ISO_STACK Inhalt, G65 schaltet die Ebenen weiter, so daß O3333.mpf einen anderen Inhalt siehr. $P_STACK zeigt weiterhin die Ebenen im Standardmode an.				
$P_STACK	**$P_IPO_STACK**	**Ebene 1**	**Ebene 2**	**Ebene 3**
1	1	O1111.mpf		
1	1	N5 G65 P2222		
2	2		O2222.mpf	
2	2		M98 P3333	
3	2			O3333.mpf
3	2			M99
2	1		M99	
Umschalten von ISO- nach Standard–Mode				
$P_STACK	**$P_IPO_STACK**	**Ebene 1**	**Ebene 2**	**Ebene 3**
1	1	O1111.mpf		
1	1	G291		
1	1	N5 M98 P2222		
2	1		O2222.mpf	
2	1		G290	
2	1		3333( )	
3	2			3333.mpf
3	2			M30
2	1		G291	
2	1		M99	
1	1	N10 M30		
Umschalten von Standard- nach ISO–Mode				
1	1	1111.mpf		
1	1	N5 G290		
1	1	N10 2222( )		
2	2		2222.mpf	
2	2		G291	
2	2		M98 P3333	
3	2			O3333.mpf
3	2			M99
2	2		G290	
2	2		M17	
1	1	N15 M30		

Liste der möglichen Unterprogramm-, Makroaufrufe im ISO–Mode

M98 Pxxxx	Unterprogrammaufruf	Ebene wird nicht verändert
M98 Pxxxx Lyyyy	Up–Aufruf mit Iteration	Ebene wird nicht verändert
G65 P	satzweiser Makro	Ebene wird erhöht
G66 P	modaler Makro	Ebene wird erhöht
G05	UP Aufruf CYCLE305	Ebene wird nicht verändert

M–Makro Subst
10814: EXTERN_M_NO_MAC_CYCLE — Ebene wird erhöht
M–Up Subst.
0715: M_NO_FCT_CYCLE — Ebene wird nicht verändert
T–Subst
10717: T_NO_FCT_CYCLE_NAME0 — Ebene wird nicht verändert
G–Subst
10816__EXTERN_G_NO_MAC_CYCLE — Ebene wird erhöht

M96 Interrupt ASUP — Ebene wird abhängig von $MC_EXTERN_FUNCTION_MASK, Bit12 verändert

Hüll–Zyklen: — Ebene wird nicht erhöht
GCode Zyklen:
G22 G23 G27 G28 G30 G30.1 G72.1 G50 — Ebene wird nicht erhöht
GCode Zyklen, Hüll–Zyklen:
$P_ISO_STACK hat für den Anwender keine Bedeutung, da kein schreibender Zugriff auf diese Zyklen möglich ist.

Abhängig von dem Maschinendatum $MC_EXTERN_FUNCTION_MASK, Bit12 wird beim Aufruf eines Interruptprogramms (ASUP) die Variable $P_ISO_STACK inkrementiert.

Bit12 = 0	beim Aufruf eines mit M96 Pxx definierten Interruptprogramms wird die Variable $P_ISO_STACK nicht verändert
Bit12 = 1	beim Aufruf eines mit M96 Pxx definierten Interruptprogramms wird die Variable $P_ISO_STACK inkrementiert

2.4 Werkzeugwechsel und Werkzeugkorrekturen

2.4.1 Werkzeugkorrekturen: T, D, M (ISO-Dialekt-M)

Werkzeugdaten, T/D-Nummer H-Nummer

Da Siemens- und ISO-Dialekt-Programme wechselweise in der Steuerung ablaufen sollen, muß die Realisierung unter Nutzung des Siemens-Werkzeugdatenspeichers erfolgen.
In jedem Korrekturspeicher der für ein Werkzeug vorhanden ist, sind die Länge, Geometrie und jeweils der Verschleiß angegeben.
Im Siemens-Mode wird der Korrekturspeicher mit T (Werkkzeugnummer) und D (Schneidennummer) angesprochen, kurz **T/D-Nummer**.
In ISO-Dialekt-M Programmen wird der Korrekturspeicher mit D (Radius) oder H (Länge), im folgenden **H-Nummer** genannt, angesprochen.
Um diese H–Nummer eindeutig einer T/D-Nummer zuzuordnen ist der Korrekturdatensatz um ein Element $TC_DPH[t,d] erweitert worden. In dieses Element wird die H-Nummer des ISO-Dialekts eingetragen.

Tabelle 2-7 Beispiel: Werkzeugkorrekturdatensatz

T	D/Schneide	H-Nummer $TC_DPH	Radius	Länge
1	1	10		
1	2	11		
1	3	12	100,00	250,00
2	1	13		
2	2	14		
2	3	15		

Beispiel:

Siemens-Programm	ISO-Dialekt-Programm
N5 T1	N5 T1
N10 G41 D3	N10 G41 H12 oder G41 D12

Bei **Programmierung des H-Wertes** im ISO-Dialekt-M Programm wird nach dem Korrektursatz mit dem passenden $TC_DPH im aktiven T gesucht und dieser angewählt.

Enthält ein Korrektudatensatz keine H-Nummer, so kann diese Korrektur im ISO-Dialekt-Mode nicht aktiviert werden.

Wird ein H programmiert, aber kein Korrektursatz mit entsprechender H-Nummer gefunden oder das dazugehörige Werkzeug T ist nicht angewählt, so erscheint ein Alarm.

2.4.2 Mögliche H–Nummern

H = 0

Alle Daten der Schneiden mit der H–Nummer 0 sind nicht verknüpft. Jede Schneide hat ihre individuellen Parameter.

H = eindeutig

Jede H-Nr. darf in einer TO-Einheit nur einmal vorkommen, sonst ist eine eindeutige Adressierung des Korrektursatzes nicht möglich. Wird trotzdem versucht, eine H-Nr. ein zweites Mal zu vergeben, wird bei einem Schreibversuch aus dem Programm ein Alarm ”17183 Kanal %1 Satz %2 H-Nummer bereits in T= %3 mit D=%4 vorhanden” abgesetzt. Dieser Alarm ist korrektursatzfähig und NC-Start-Clear.

Beispiel:

```
N5  $TC_DPH[1,1] = 5
N10 $TC_DPH[2,1] = 5
```

Wird versucht über BTSS (HMI, PLC) eine H–Nummer zweimal zu vergeben, wird dieser Schreibversuch negativ quittiert,

Ändern des Korrekturspeichers

Bestehende Werkzeugkorrekturen können mit G10 überschrieben werden. Neue Werkzeugkorrekturen werden mit G10 nicht angelegt.

Längenkorrektur Geometrie:	G10 L10 Pxx Ryy
Längenkorrektur Verschleiß:	G10 L11 Pxx Ryy
Radiuskorrektur Geometrie:	G10 L12 Pxx Ryy
Radiuskorrektur Verschleiß:	G10 L13 Pxx Ryy

P gibt die H–Nummer des Korrekturspeichers und R den Wert an.
Anstelle von L11 kann L1 programmiert werden.

Aktive Ebene

Damit die Zuordnung der Werkzeuglängenkorrekturen zu den Geometrieachsen unabhängig von der Ebenenanwahl ist, muß das Settingdatum 42940: $SC_TOOL_LENGTH_CONST den Wert 17 enthalten. Damit ist Länge 1 immer der Z-Achse zugeordnet.

Anwahl der Werkzeuglänge

Die Werkzeuglänge und der Werkzeugradius werden immer mit D oder H programmiert.

Beispiel:

T	D/Schneide	H-Nummer $TC_DPH	Radius	Länge
2	3	4	10	15

Iso-Dialekt-M:
T2
G43 H4 oder D4 ;Längenanwahl
G42 D4 oder H4 ;Radiusanwahl

Bei Iso- Dialekt-M Programmen, die mit unterschiedlichen D- und H-Nummern programmiert sind, muß der Korrekturwert 2x eigetragen werden.

Beispiel:

T	D/Schneide	H-Nummer $TC_DPH	Radius	Länge
2	3	4	10	15
2	4	5	10	15

Iso-Dialekt-M:
T2
G43 H4 ;Längenkorrektur aus T2 D3
G42 D5 ;Radius- und Längenkorrektur aus T2 D4

Flache D-Nummer

Sind flache D–Nummern aktiv, wird das T unabhängig von der H–Nummer programmiert. Die Überprüfung, ob die H–Nummer zum angewählten Werkzeug passt, findet nicht mehr statt.

Auch bei flachen D-Nummern muß jedem Korrekturspeicher eine H-Nummer zugeordnet werden.

Werkzeugverwaltung

Bei aktiver Werkzeugverwaltung haben Schwesterwerkzeuge dieselbe H-Nummer. Die Unterscheidung erfolgt über Duplo-Nummern
Mit H99 wird bei aktiver **Werkzeugverwaltung** die Korrektur D1 zum aktuell angewählten Werkzeug aktiviert.

Im ISO-Dialekt-M sind als Werkzeugbezeichner nur numerische Ausdrücke möglich. Strings als Bezeichner sind nicht zulässig.
Beispiel: T = ”2”, Anwahl mit T2.

Längenkorrektur in mehreren Achsen

Längenkorrekturen können in mehreren Achsen aktiviert werden. Die Anzeige der resultierenden Längenkorrektur ist allerdings nicht möglich.

In der Anzeige für aktive T- und D-Nummer erscheinen die Siemens T- und D-Nummer.
Für die aktive ISO-Dialekt H- und D-Nummer gibt es neue BTSS-Variablen, die angezeigt werden können.

Über das Maschinendatum 22220: $MC_AUXFU_T_SYNC_TYPE wird festgelegt, ob die Ausgabe an die PLC während oder nach der Bewegung ausgeführt wird.
Über das Maschinendatum 20110: $MC_RESET_MODE_MASK, Bit6 kann die Wirksamkeit der Werkzeuglängenkorrektur über Reset hinaus eingeschaltet werden.

Beispiel: Werkzeuganwahl ISO-Dialekt-M:

```
;(Fanuc 0 M Werkzeugkorrektur mit T, Schneidennummer
;(die Korrekturen werden)
;(mit G10 geschrieben werden)
G290

;Werkzeugkorrekturspeicher T2 Schneide1:
N5000 $TC_DP1[2,1]=10       ;Typ
N5000 $ TC_DP1[2,1]=7       ;ISO H Nummer

;Werkzeugkorrekturspeicher T3 Schneide2:
N5000 $TC_DP1[3,2]=10       ;Typ
5000 $TC_DP1[3,2]=3         ;ISO H Nummer

;Werkzeugkorrekturspeicher T4 Schneide3:
N5000 $TC_DP1[4,3]=10       ;Typ

N5000 $TC_DP1[4,3]=8        ;ISO H Nummer

G291

;Werkzeugkorrekturen schreiben
;------------------------
;T2 Schneide1
G10 L12 P7 R5
;T3 Schneide 2
G10 L10 P3 R15
G10 L12 P3 R10

N8      G01 G40 F5000 X0 Y0 Z0
N10     X50.
N15     50
N17     Z10.
N20     X0
N25     Y0
N30     X-10 Y-10
```

```
N30    T2                   ;Werkzeug 2
N33    G43 H7 Z0            ;H-Nummer 7
N35    G41 X0 Y0 Z0 D7
N40    X50.
N45    Y50.
N48    Z10.
N50    X0
N55    Y0
N60    G40 X-10 Y-10

N65    T3
N68    G43 H3 Z0
N70    G42 X0 Y0 Z0 D3
N75    X50.
N77    Y50.
N78    Z10.
N80    X0
N85    Y0
N90    G40 X-10 Y-10

N95    T4
N98    G43 H8 Z0
N100   G41 X0 Y0 Z0 D8
N105   X50.
N110   Y50.
N112   Z10.
N115   X0
N120   Y0
N125   G40 X-10 Y-10

M30
```

Ob die Korrektur im Anwahlsatz oder erst bei der nächsten Programmierung der Achse erfolgt, wird über das Maschinendatum
20382: $MC_TOOL_CORR_MOVE_MODE festgelegt.

2.4.3 Werkzeugkorrektur: T (ISO-Dialekt-T)

Die Werkzeugdaten liegen in dem Siemens-Werkzeugdatenspeicher.
Jedes Werkzeug besteht aus vier Einträgen, jeweils für die X, Z Achse und Radius sowie Schneidenlage. Der Wertebereich für Längen- und Radiuskorrektur beträgt ± 999.999 mm. Für die Schneidenlage gilt der Wertebereich 0 - 9, wobei 0 und 9 identisch sind.
Die Bedeutung entspricht der Schneidenlage bei Siemens-Drehwerkzeugen.

Die Anwahl erfolgt mit T. Dabei enthält T die Werkzeug- und die Korrekturnummer.

Es wird die Korrektur entweder mit der Siemens Schneidennummer D oder mit der H–Nummer aus $TC_DPH adressiert. Die Adressierung mit D ist nur bei ”flachen D–Nummern” möglich. Wird die Werkzeugverwaltung genutzt, erfolgt die Adressierung immer mit H.

Txxxxyyyy: xxxx=Werkzeugnummer, yyyy=Korrekturnummer

Das Maschinendatum 10890: $MN_EXTERN_TOOLPROG_MODE, Bit 0 legt fest, wie der T-Wert interpretiert wird.

Im Maschinendatum 10888: $MN_EXTERN_DIGITS_TOOL_NO wird die Anzahl der Stellen der Werkzeugnummer festgelegt. Die Stellen werden von links gezählt. Die nachfolgenden Stellen geben die Korrekturnummer an.

Mit Bit 0=1 im MD 10890 wird die Korrekturnummer gleich der Werkzeugnummer gesetzt.

Beispiel:

```
$MN_EXTERN_TOOLPROG_MODE=0
$MN_EXTERN_DIGITS_TOOL_NO=2
T1234        ;Hilfsfunktion T1234 an PLC
             ;Werkzeugnummer 12
             ;Korrekturanwahl D34/H34

T123         ;Hilfsfunktion T123 an PLC
             ;Werkzeugnummer 12
             ;Korrekturanwahl D3/H3

$MN_EXTERN_TOOLPROG_MODE, Bit0=1
T12          ;Hilfsfunktion T12 an PLC
             ;Werkzeugnummer 12
             ;Korrekturanwahl 12
```

Über das Maschinendatum 20382: $MC_TOOL_CORR_MOVE_MODE wird ausgewählt, wann die Korrektur herausgefahren wird: Direkt im Anwahlsatz oder erst beim Programmieren der Achse.

Über MD 20110: $MC_RESET_MODE_MASK, Bit 6 wird bestimmt, ob die Korrektur mit Reset erhalten bleibt oder abgewählt wird.

Mit MD 20360: $MC_TOOL_PARAMETER_DEF_MASK, Bit 0 wird eingestellt, daß der Verschleißwert der Planachse als Durchmesserwert gerechnet wird. Die Korrektur Geometrie wirkt immer als Radius.

Beispiel: Werkzeuganwahl ISO-Dialekt-T:

```
G290
N5000 ;Definition des Werkzeugkorrekturspeichers D1:
N5000 $TC_DP1[1,1]=10       ;Typ
N5000 $TC_DP2[1,1]=9        ;Schneidenlage
N5000 $TC_DP6[1,1]=5        ;Radius
N5000 ;Definition des Werkzeugkorrekturspeichers D2:
N5000 $TC_DP1[2,1]=10       ;Typ
N5000 $TC_DP2[2,1]=1        ;Schneidenlage
```

```
N5000   $TC_DP6[2,1]=5          ;Radius

G291
;Werkzeugkorrekturdaten schreiben
N3      G10 P1 X10 Z20 Y30
N5      G10 P2 X20 Y20 Z100

N10     G00 G18 X0 Y0 Z0
N10 T0101                       ;Werkzeug 1, Schneide 1
N15     G00 X10 Y10 Z10
N20     T0201                   ;Werkzeug 1, Schneide 1
N25     G00 X10 Y10 Z10
...
M30
```

Ändern des Korrekturspeichers

Bestehende Werkzeugkorrekturen können mit G10 überschrieben werden. Neue Werkzeugkorrekturen werden mit G10 nicht angelegt.

G10 P<100 / 10000 X Y R Q   Geometrie
G10 P>100 / 10000 X Y R Q   Verschleiß

P100/10000	;Mit MD 20734: EXTERN_FUNCTION_MASK, Bit1 wird ausgewählt, ;ob bei P<100 oder 10000 nach Geometrie oder Verschleiß ;unterschieden wird.
X Y Z	;Korrekturwerte absolut oder inkrementell, abhängig von G90/91
U V W	;Korrekturwerte inkrementell
R	;Radius
Q	;Schneidenlage

Werkzeugkorrekturanwahl mit $TC_DPH

Bisher ist bei ISO–Dialekt–T immer die Funktion “flache D–Nummer” aktiv. Hierbei sind die D–Nummern eindeutig, mit dem Befehl Txxyy bzw. G10 Pyy wird mit yy die Siemens Schneidennummer adressiert. Um die Werkzeugverwaltung zu benutzen, ist es notwendig, im ISO–Dialekt–T strukturierte D–Nummern zu adressieren. Daher erhält, genauso wie im ISO–Dialekt–M, jede Schneide einen Paramter $TC_DPH[ ] der es erlaubt, eine Schneide eindeutig innerhalb einer TO–Einheit zu adressieren.

Eingeschaltet wird die Funktion mit dem MD 10890:
$MN_EXTERN_TOOLPROG_MODE Bit 2=1.
Ist die Funktion aktiv, muß bei ISO–Dialekt–T die Werkzeugkorrektur immer mit der H–Nummer adressiert werden. Programme, die die Schneidennummer adressieren laufen nicht mehr. Der Paramter $TC_DPH[ ] wird nur angelegt, wenn $MN_EXTERN_TOOLPROG_MODE Bit 2=1 ist. Die H–Nummern müssen innerhalb einer TO–Einheit eindeutig vergeben werden, sonst kommt es zu Alarm 17183.

Es gibt 3 Möglichkeiten:

1. **flache D–Nummer + $MN_EXTERN_TOOLPROG_MODE Bit 2=0**
   Es wird die Korrektur immer mit der Schneide D adressiert.

```
G290
N605 $TC_DP1[1]= 500
N615 $TC_DP1[2]= 500
N625 $TC_DP1[3]= 500
N635 $TC_DP1[4]= 500

G291
N650 G10 P2   X10          ; Geometrie Schneide 2 schreiben
N655 G10 P102 X1           ; Verschleiß Schneide 2 schreiben

N670 T0102     ;Schneide 2 anwählen
N675 T0105     ;Alarm , weil Schneide 5 nicht vorhanden ist.
```

2. **flache D–Nummer + $MN_EXTERN_TOOLPROG_MODE Bit 2=1**
   Es wird die Korrektur immer mit der H–Nummer adressiert.

```
G290
N705 $TC_DP1[1]= 500
N708 STC_DPH[1]=11

N710 $TC_DP1[2]= 500
N715 STC_DPH[2]=22
N720 $TC_DP1[3]= 500
N725 STC_DPH[3]=33
N730 $TC_DP1[4]= 500
N735 STC_DPH[4]=44

G291
N740 G10 P22   X10.        ; Geometrie Schneide 2 schreiben
N745 G10 P122 X1.          ; Verschleiß Schneide 2 schreiben
N747 G10 P55   X10.        ; Alarm 12550, weil Schneide mit H55 nicht vo
                           ; handen ist.

N750 T0122     ; Schneide 2 wird angewählt
N752 T0155     ; Alarm 12550, weil Schneide mit H55 nicht vorhanden
               ; ist.
```

3. **strukturierte D–Nummer + $MN_EXTERN_TOOLPROG_MODE Bit 2=1**
   Es wird die Korrektur immer mit der H–Nummer adressiert.

```
G290
N805 $TC_DP1[1,1]= 500
N808 STC_DPH[1,1]=11

N810 $TC_DP1[1,2]= 500
N815 STC_DPH[1.2]=22

N820 $TC_DP1[2,1]= 500
N825 STC_DPH[2,1]=33
```

```
N830 $TC_DP1[2,2]= 500
N835 STC_DPH[2,2]=44

G291
N840 G10 P22   X10      ; Geometrie T1 Schneide 2 schreiben.
N845 G10 P122 X1        ; Verschleiß T1 Schneide 2 schreiben.
N847 G10 P55 X1         ; Alarm 12550, weil Schneide mit H55 nicht vorha
                        ; den ist.

N850 T0122     ;T1 Schneide 2 anwählen.
N855 T0244     ;T2 Schneide 2 anwählen. Alarm , weil in T2 keine Schneide mit
               ; H22 vorhanden ist.
```

2.4.4 Werkzeugwechselzyklus

ISO-Dialekt-Mode

Über das MD 10717: T_NO_FCT_CYCLE_NAME wird dem T-Befehl ein Unterprogramm zugeordnet. Jeder Satz, der einen T-Befehl enthält, wird erst ausgeführt, und anschließend wird das Unterprogramm aufgerufen. Eine Ausgabe des T-Wertes erfolgt nicht, das T-Wort muß im Zylus noch einmal programmiert werden.

Im Unterprogramm kann über die Systemvariable $C_T_PROG bzw. $C_D_PROG abgefragt werden, ob T bzw. D programmiert wurde. Mit den Systemvariablen $C_T bzw. $C_D können die Werte ausgelesen werden. Wird im Unterprogramm erneut ein T-Befehl programmiert, wird keine Ersetzung ausgeführt, sondern das T-Wort an die PLC ausgegeben.

Über die MD 10715: M_NO_FCT_CYCLE und 10716: M_NO_FCT_CYCLE_NAME kann einer M-Funktion (z.B. M06) ein Unterprogramm zugewiesen werden.

Die Abbildung der M-, T-Programmierung auf Zyklenaufrufe wirkt im ISO-Dialekt-Mode wie im Siemens-Mode.

Sind T und M6 in einem Satz programmiert, so kann in dem durch M6 aufgerufenen Zyklus bereits die programmierte T-Nummer über $P_TOOL abgefragt werden. Im Satzsuchlauf wird der M-/T-Aufruf ebenfalls auf den Zyklenaufruf abgebildet. Ein Start des Wechselzyklus nach Suchlauf-Ende muß von PLC-Seite gelöst werden.

Ablauf:

```
N20 T1234
N30 M6        ;Werkzeug wechseln
N40 H3 G43    ;Längenkorrektur aktivieren in T1234
N50 T333      ;Werkzeug-Vorwahl
N60 G1 X10    ;Korrektur T1234 ist aktiv
N70 M6        ;Werkzeug 333 einwechseln, D0 H0 aktiv
N80 H4        ;neue Längenkorrektur aktivieren
N90 .....
```

■

Zyklen und Konturzug 3

3.1 Ablauf von Zyklenaufrufen im externen CNC–System über G–Befehle

Allgemeine Beschreibung

Die Funktionalität der ISO–Dialekt–Zyklen wird in den Siemens–Standardzyklen realisiert:
Aus dem ISO–Dialekt–Programm wird ein Hüllzyklus aufgerufen. Alle im Satz programmierten Adressen werden diesem Hüllzyklus in Form von Systemvariablen übergeben. Der Hüllzyklus paßt diese Daten dem Siemens–Standardzyklus an und ruft dessen Namen auf.
Der Maschinenhersteller hat die Möglichkeit, die Hüllzyklen durch eigene Zyklen zu ersetzen.

Zyklenparameter

Bei den Bearbeitungszyklen sind verschiedene Zyklenparameter in kanalspezifischen GUDs (Globale User Daten) vorzubesetzen. Die Namen und die Bedeutung der GUDs sind im Kapitel 3.2 aufgelistet.

Ablauf bei Zyklenaufruf über G-Befehl

Bild 3-1 Allgemeiner Zyklenablauf im ISO-Dialekt_Mode

Hüllzyklus

Im Hüllzyklus werden die Anpassungen, welche aufgrund der ISO–Dialekt–Programmiersyntax notwendig sind, vorgenommen. Dadurch müssen die bestehenden SINUMERIK–Zyklen nicht verändert werden. Der Name des Hüllzyklus ist fest vorgegeben.

Ablauf:

4. Im ISO–Dialekt–Mode ist der Zyklus (z.B. G81) programmiert
5. Es erfolgt eine automatische Umschaltung in den Siemens–Mode und der Aufruf des dazugehörigen Hüllzyklus (siehe Bild LEERER MERKER)
6. Der Hüllzyklus ruft den dazugehörigen Siemens–Standardzyklus auf

Eine Programmierung mit G290 ist nicht notwendig. Beim Rücksprung wird automatisch in das externe CNC–System gewechselt.

Achtung

Die Zyklen dürfen nur mit den G–Befehlen aufgerufen werden.
Dadurch wird sichergestellt, daß der Hüllzyklus mit den entsprechenden Zyklenparametern versorgt wird.

Der Hüllzyklus **darf nicht** direkt mit CALL CYCLE3xx aufgerufen werden!

Modal wirksame Zyklen

Ist ein modaler Zyklus aktiv, wird in jedem NC–Satz der Hüllzyklus aufgerufen. Sind im NC–Satz keine Achspositionen (X, Y oder Z) programmiert, wird der Siemens–Standardzyklus nicht aufgerufen.

Im Satz programmierte Adressen (F usw.) werden durch den Hüllzyklus aktiviert. Wurde z.B. kein Vorschub programmiert, wird der aktuelle Vorschub als Bahnvorschub übernommen.

Während ein modaler Zyklus aktiv ist, können in den Folgesätzen Zyklenparameter programmiert werden. Diese Parameter werden in die Systemvariablen geschrieben, damit der Hüllzyklus mit den geänderten Parametern arbeitet.
Modale Zyklen werden, anders als modale Makros, bereits im Aufrufsatz (z.B. Satz mit G81 usw.) ausgeführt.

Abwahl des Zyklus:
Die Abwahl erfolgt mit G80 oder mit einer G–Funktion der 1. G–Gruppe.

Beispiel:

```
N10 G81 X10. Y20. Z–15. R5 F1000
                           Bohrposition X10 mm, Y20 mm
                           Bohrtiefe Z–15 mm
                           Referenzebene 5 mm
                           Bohrvorschub F.. (mm/min oder mm/U)
N20 X50. Y30. R10          Bohrposition X50 mm, Y30 mm,
                           neue Referenzebene 10 mm
N30 G80                    Zyklus G81 löschen
```

Zyklenvarible schreiben, abhängig von programmierten Adressen im Satz

Bei aktiven modalen Zyklen, wurden bisher immer alle programmierten Adressen im Satz in die Zyklenvariablen geschrieben. Die Auswertung der Variablen erfolgt im Zyklus, dort wird aufgrund der Zyklenlogik entschieden, wie die Variablen verwendet werden müssen.

In manchen Fällen führt dies dazu, daß die Zyklenparameter auch dann beschrieben werden, wenn sie aufgrund der Programmiersyntax nicht als Zyklenparameter interpretiert werden dürfen.

Bei folgenden Funktionen werden deshalb keine oder nicht mehr alle programmierten Adressen in die Zyklenparameter geschrieben:

M98 P3 L2 X10 Y20	Die Adressen Pxx und Lxx werden nicht in die Zyklenparameter geschrieben.
G05 P5 L2 X10 Y20	Die Adressen Pxx und Lxx werden nicht in die Zyklenparameter geschrieben.
G05 P1 L2 X10 Y20	Bei aktivem modalem Zyklus wird der Alarm 12722 ausgegeben, da hier nach dem modalen Zyklus CYCLE305 aufgerufen wird, für den die progammierten Werte eigendlich gedacht sind
G54 P10 X10 Y20 M44	Die Adresse Pxx wird nicht in die Zyklenparameter geschrieben
G31 P98 X30 F100	Die Adressen Pxx, Fxx und Achswerte werden nicht in die Zyklenparameter geschrieben
G31 P1 X30 Y20 F100	Keine der programmierten Adressen wird in die Zyklenparameter geschrieben
G51 P1000 I2 J3 K2 X30 Y40	Keine der programmierten Adressen wird in die Zyklenparameter geschrieben
G50 P10000 X10 Y30	Alle Parameter werden in die Zyklenparameter geschrieben

3.2 Globale User Daten (GUD)

Tabelle 3-1 GUD7 für programmierte Zyklenwerte (ISO–Dialekt–Programmdaten)

GUD	Beschreibung/Verwendung	CYCLE
Realwerte		
_ZFPR[0]	Ausgangsebene (aktuelle Position bei 1. Aufruf mit G..), Rückzugsposition bei G98 aktiv	381M, 383M, 384M, 387M
_ZFPR[1]	Referenzebene, Rückzugsposition bei G99 aktiv (bei G87 Rückzug nur auf Ausgangsposition möglich).	381M, 383M, 384M, 387M
_ZFPR[2]	Endbohrtiefe, absolut	381M, 383M, 384M, 387M
_ZFPR[3]	Rückzugsposition, abhängig von G98/G99 (Ausgangsebene/ R–Ebene)	381M, 383M, 384M, 387M
_ZFPR[4]	Bohrvorschub	381M, 383M, 384M, 387M
_ZFPR[5]	Verweilzeit (s) auf Endtiefe (G82/G89/G76/G87)	381M, 384M, 387M
_ZFPR[6]	1. Bohrtiefe (Einzelbohrtiefe) inkr. (G73/G83)	383M
_ZFPR[7]	1. Bohrtiefe absolut (G73/G83)	383M
_ZFPR[8]	Abhebe–/Zustellweg (G76)	387M
_ZFPR[9]	Drehzahl für Gewindebohren (G74/G84)	384M
_ZFPR[20]	Ausgangsebene (aktuelle Position bei 1. Aufruf)	383T, 384T, 385T
_ZFPR[21]	R–Ebene	383T, 384T, 385T
_ZFPR[22]	Endbohrtiefe, absolut	383T, 384T, 385T
_ZFPR[23]	Rückzugposition (1=G98, 2=G99)	383T, 384T, 385T
_ZFPR[24]	Gewindesteigung/Bohrvorschub	376T, 383T, 384T, 385T
_ZFPR[25]	Verweilzeit auf Endtiefe	383T, 384T, 385T
_ZFPR[26]	Drehzahl zum Gewindebohren	384T
_ZFPR[27]	Endpunkt X	371T, 372T, 373T, 376T
_ZFPR[28]	Endpunkt Z	371T, 372T, 373T, 376T
_ZFPR[29]	Anfangspunktversatz X (Kegelgewinde)	371T, 372T, 376T
_ZFPR[30]	Gewindestartpunkt X	376T
_ZFPR[31]	Gewindestartpunkt Z	376T
_ZFPR[32]	erste Bohrtiefe	383T
Integerwerte		
_ZFPI[0]	Aktueller G–Code des ISO–Dialektes–Bohrzyklus	381M, 383M, 384M

Tabelle 3-1 GUD7 für programmierte Zyklenwerte (ISO–Dialekt–Programmdaten)

GUD	Beschreibung/Verwendung	CYCLE
_ZFPI[1]	M–Funktion für Spindel Start (M3, M4) nach Spindel Stop	381M, 384M
_ZFPI[20]	aktueller G–Code des Gewindezyklus/Bohrzyklus	383T, 384T, 385T
_ZFPI[21]	Spindelrichtung (3=M3, 4=M4)	383T, 384T, 385T
_ZFPI[22]	Abspanbearbeitungsart Schruppen	370T, 371T, 372T, 373T
_ZFPI[23]	Bearbeitungsart Tiefloch/Bohren	383T

Tabelle 3-2 GUD7 für Zyklen–Settingdaten (ISO–Dialekt–Settingdaten)

GUD	Beschreibung/Verwendung	
Realwerte		
_ZSFR[0]	Sicherheitsabstand zur Referenzebene	381M, 383M
_ZSFR[1]	Rückzugsbetrag zum Späne brechen (G73)	383M
_ZSFR[2]	Winkelversatz für orientierten Spindelstop, Werkzeug muß in Richtung +X ausgerichtet sein (G76) Rückzugsrichtung: –X G17 Ebene XY –Z G18 Ebene ZX –Y G19 Ebene YZ	387M
_ZSFR[20]	Sicherheitsabstand zur Referenzebene	383T, 384T
_ZSFR[21]	Sicherheitsabstand zum Spänebrechen	383T, 385T
Integerwerte		
_ZSFI[0]	0=Bohrachse ist senkrecht zur Ebene (Standard) 1=Bohrachse immer ”Z”	381M, 383M, 384M,387M
_ZSFI[1]	0= Gewindebohren ohne Ausgleichsfutter 1= Gewindebohren mit Ausgleichsfutter 2= Tieflochgewindebohren mit Späne brechen 3= Tieflochgewindebohren mit Späne entfernen	384M,387M
_ZSFI[2]	Faktor (1–200%) für Rückzugsdrehzahl bei Gewindebohren (G74/G84)	384M
_ZSFI[3]	Polarkoordinaten 0=AUS 1=EIN	381M, 383M, 384M,387M
_ZSFI[20]	Tieflochbohren mit Spände brechen/entfernen	383T, 385T
_ZSFI[22]	Faktor für Rückzugsdrehzahl	384T
_ZSFI[23]	Verweilzeit bei G95, 0=Sekunden, 1=Umdrehungen	383T
_ZSFI[24]	Anzahl Leerschnitte	376T
_ZSFI[25]	Schneidenwinkel	376T
_ZSFI[26]	Gewindeauslaufweg (n*Steigung)	376T
_ZSFI[27]	min. Zustelltiefe	376T
_ZSFI[28]	Schlichtaufmaß	376T
_ZSFI[29]	Wegfahrbetrag bei Einstichzyklus	374T
_ZSFI[30]	Schnitttiefe bei Abspanzyklus	371T, 372T
_ZSFI[31]	Wegfahrbetrag bei Abspanzyklus	371T, 372T

Tabelle 3-2 GUD7 für Zyklen–Settingdaten (ISO–Dialekt–Settingdaten)

GUD	Beschreibung/Verwendung	
_ZSFI[32]	Zustellwert X–Achse bei Konturwiederholung	373T
_ZSFI[33]	Zustellwert Z–Achse bei Konturwiederholung	373T
_ZSFI[34]	Teilungszahl bei Konturwiederholung	373T
_ZSFI[39]	G–Code–System 2=B, 1=A, 3=C	300, 328, 330, 370T, 371T, 372T, 373T, 374T, 376T

3.3 Bohrzyklen (ISO–Dialekt–M)

3.3.1 Übersicht und Parameterbeschreibung

Die Bohrzyklen sind modal wirksam. Während ein Bohrmodus aktiv ist, müssen bei Parameteränderung nur die neuen Parameter programmiert werden.

Es erfolgt **keine** Verfahrbewegung, wenn:

- im NC–Satz kein Wert für X, Y und Z programmiert ist
- die Anzahl der Wiederholungen K=0 programmiert wurde

Die Rückzugsposition gilt für alle Bohrzyklen

- G98 Rückzug auf Ausgangsebene
- G99 Rückzug auf Referenzebene

Übersicht

Tabelle 3-3 Übersicht der Bohrzyklen

externer Zyklusaufruf	Beschreibung
G73 X.. Y.. Z.. R.. F.. Q..	Tieflochbohrzyklus mit Späne brechen
G74 X.. Y.. Z.. R.. F.. P..	Gewindebohrzyklus links
G76 X.. Y.. Z.. R.. F.. Q.. P..	Feinbohrzyklus
G80	Zyklus aus; Zyklus wird auch durch Programmierung einer G–Funktion aus der 1. G–Gruppe abgewählt.
G81 X.. Y.. Z.. R.. F..	Bohrzyklus; Bohren, Rückzug mit G00
G82 X.. Y.. Z.. R.. F.. P..	Bohrzyklus; Bohren, Verweilzeit, Rückzug mit G00
G83 X.. Y.. Z.. R.. F.. Q..	Tieflochbohrzyklus mit Späne entfernen
G84 X.. Y.. Z.. R.. F.. P..	Gewindebohrzyklus rechts
G85 X.. Y.. Z.. R.. F..	Bohrzyklus; Bohren, Rückzug mit Bohrvorschub
G86 X.. Y.. Z.. F.. R.. K..	Bohrzyklus, Rückzug mit G00
G87 X.. Y.. Z.. F.. R.. P.. Q.. K..	Rückwärtssenken
G89 X.. Y.. Z.. F.. R.. P.. K..	Bohrzyklus, Rückzug mit Arbeitsvorschub

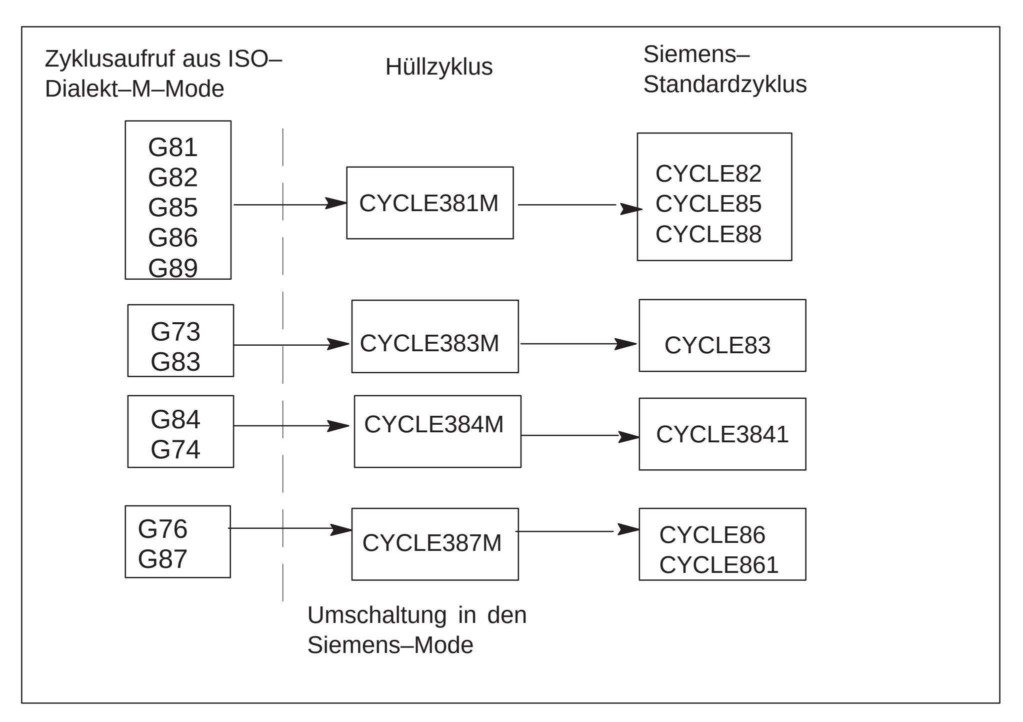


Bild 3-2 Zuordnung des Zyklenaufrufs im ISO-Dialekt-M-Mode über Hüllzyklus zum Siemens-Standardzyklus

Beispiel: ISO–Dialekt–M

```
N10 G81 X100. Z–50. R20  F100
```

Durch G81 wird automatisch der Hüllzyklus CYCLE381M aufgerufen.
Im Hüllzyklus werden die Berechnungen durchgeführt und anschließend der Standard–Bohrzyklus CYCLE81 aufgerufen.

Parameterbeschreibung

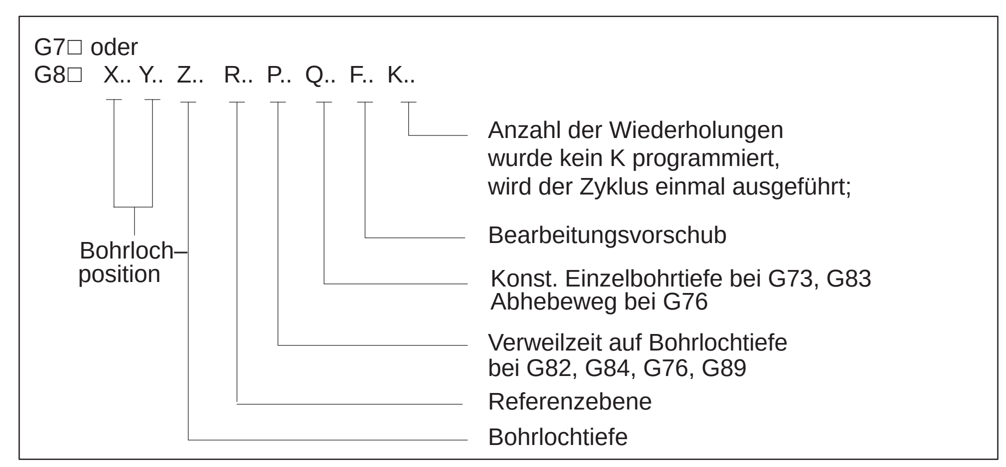


Bild 3-3 Beschreibung der zulässigen Parameter bei G17 (Ebene x/Y)

Ebene

Tabelle 3-4 Definition der Ebene

definierte Ebene	Position der Bohrung	Tiefe
G17	X, Y	Z
G18	Z, X	Y
G19	Y, Z	X

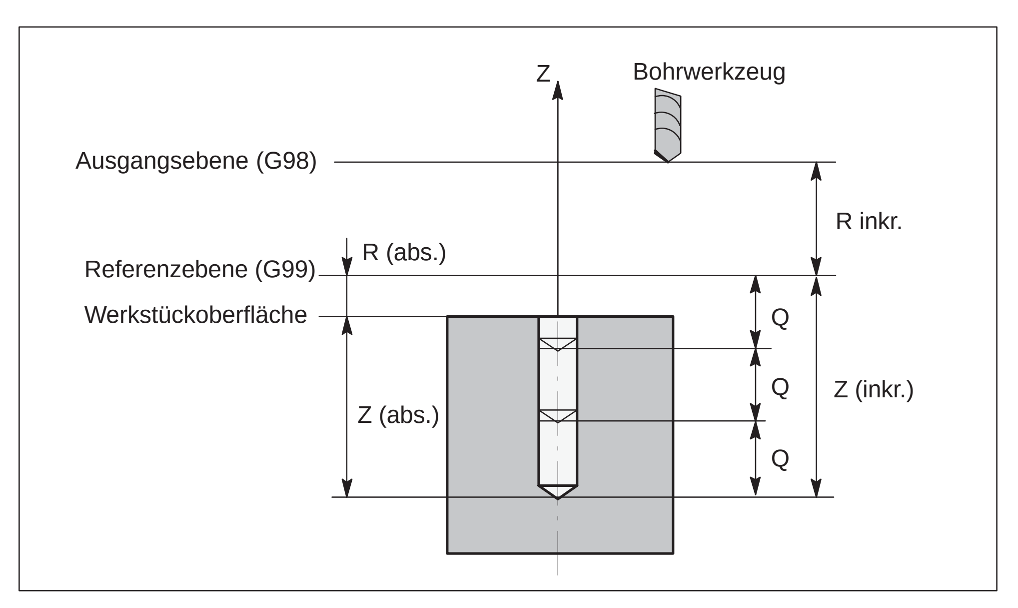


Bild 3-4 Beispiel Tieflochbohrzyklus G83 mit definierter Ebene G17
Darstellung der Ausgangs- und Referenzebene sowie Parameter

3.3.2 Beschreibung Hüllzyklus CYCLE381M

Der Aufruf erfolgt im ISO–Dialekt–M–Mode über die G–Befehle G81, G82, G85.

Hinweise

Die Bohrachse ist über GUD _ZSFI[0] zu definieren (siehe Kapitel 3.2).

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein.

Über GUD _ZSFR[0] kann ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der Referenzebene berücksichtigt, muß bei _ZSFR[0] der Wert 0 eingetragen werden.

Wurde keine Referenzebene programmiert, erfolgt das Bohren von der Ausgangsebene (aktuelle Position) aus.

Die Endbohrtiefe muß programmiert sein. Andernfalls wird ein Alarm angezeigt.

Ist kein Vorschub programmiert, wird der aktuelle Vorschub als Bohrvorschub übernommen.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.3.3 Beschreibung Hüllzyklus CYCLE383M

Der Aufruf erfolgt im ISO–Dialekt–M–Mode über die G–Befehle G73, G83.

Hinweise

Die Bohrachse ist über GUD _ZSFI[0] zu definieren (siehe Kapitel 3.2).

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein.

Über GUD _ZSFR[0] kann ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der Referenzebene berücksichtigt, muß bei _ZSFR[0] der Wert 0 eingetragen werden.

Wurde keine Referenzebene programmiert, erfolgt das Bohren in Schritten von der Ausgangsebene (aktuelle Position) aus.

Die Endbohrtiefe und die Einzelbohrtiefe Q muß programmiert sein. Andernfalls wird ein Alarm angezeigt.

Ist kein Vorschub programmiert, wird der aktuelle Vorschub als Bohrvorschub übernommen.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

Tieflochbohren mit Späne entfernen

Im Siemens–Standardzyklus wird der Vorhalteabstand über eine Formel ermittelt, d.h. er kann nicht frei gewählt werden.

Zum Entspänen wird auf Referenzebene zurückgezogen.

Einzelbohrtiefe ”Q”:

– Wenn ”Q” fehlt oder Q ≤ 0, wird ein Alarm angezeigt.

– Ist Q > Gesamttiefe, erfolgt ein Bohrvorgang bis zur Endtiefe.

– Ist Q > Gesamttiefe/2 erfolgt der 1. Bohrvorgang mit dem Wert von Q. Der Rest wird dann mit einem Bohrvorgang gebohrt.

– Ist Q < Gesamttiefe/2 wird die Einzelbohrtiefe solange ausgeführt, bis die Resttiefe < Q/2 ist. Der Rest wird dann in 2 gleichgroße Zustellungen aufgeteilt.

Tieflochbohren mit Späne brechen

Der Rückzugsbetrag zum Späne brechen wird über GUD _ZSFR[1] definiert.

– _ZSFR[1] > 0 Rückzugsbetrag wie eingegeben

– _ZSFR[1] ≤ 0 Rückzugsbetrag ist immer 1 mm

Einzelbohrtiefe ”Q”:

– Wenn ”Q” fehlt oder Q ≤ 0, wird ein Alarm angezeigt.

– Ist Q > Gesamttiefe, erfolgt ein Bohrvorgang bis zur Endtiefe.

– Ist Q < Gesamttiefe wird die Einzelbohrtiefe solange ausgeführt, bis die Resttiefe ≤ Q ist. Der Rest wird dann in einem Bohrvorgang gebohrt.

3.3.4 Beschreibung Hüllzyklus CYCLE384M

Der Aufruf erfolgt im ISO–Dialekt–M–Mode über die G–Befehle G74, G84.

Hinweise

Die Bohrachse ist über GUD _ZSFI[0] zu definieren (siehe Kapitel 3.2).

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein.

Über GUD _ZSFR[0] kann ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der Referenzebene berücksichtigt, muß bei _ZSFR[0] der Wert 0 eingetragen werden.

Über GUD _ZSFI[2] kann die Drehzahl beim Rückzug beeinflußt werden.
Beispiel: _ZSFI[2]=80, der Rückzug erfolgt mit 80% der Bohrdrehzahl.

Wurde keine Referenzebene programmiert, erfolgt das Bohren von der Ausgangsebene (aktuelle Position) aus.

Die Endbohrtiefe muß programmiert sein. Andernfalls wird ein Alarm angezeigt.

Ist kein Vorschub programmiert, wird der aktuelle Vorschub als Bohrvorschub übernommen.

Wurde der Bohrvorschub mit mm/min (Inch/min) angegeben, wird der programmierte Vorschubwert in Abhängigkeit der zuletzt programmierten Drehzahl in den entsprechenden Umdrehungsvorschub umgerechnet und als Steigungswert an den Standard–Gewindebohrzyklus CYCLE84 übergeben.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.3.5 Beschreibung Hüllzyklus CYCLE387M

Der Aufruf erfolgt im ISO–Dialekt–M–Mode über den G–Befehl G76 und G87.

Hinweise

Die Bohrachse ist über GUD _ZSFI[0] zu definieren (siehe Kapitel 3.2)

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein.

Über GUD _ZSFR[0] kann ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der Referenzebene berücksichtigt, muß bei _ZSFR[0] der Wert 0 eingetragen werden.

Der Abhebeweg wird immer auf die negative Richtung der ersten Geometrieachse bezogen

bei Ebene G17	Abhebeweg in –X
bei Ebene G18	Abhebeweg in –Z
bei Ebene G19	Abhebeweg in –Y

Der Winkel muß deshalb so eingegeben werden, daß die Werkzeugspitze in der definierten Ebene nach dem Spindelstop in die Plus–Richtung (+) zeigt.
Beispiel: Ebene G17 aktiv, dann muß Werkzeugspitze in Richtung +X zeigen.

Wurde keine Referenzebene programmiert, erfolgt das Bohren von der Ausgangsebene (aktuelle Position) aus.

Die Endbohrtiefe muß programmiert sein. Andernfalls wird ein Alarm angezeigt.

Ist kein Vorschub programmiert, wird der aktuelle Vorschub als Bohrvorschub übernommen.

Ist kein Abhebebetrag programmiert, wird Q = 0 gesetzt. Der Zyklus wird dann ohne Abheben ausgeführt.

Nach dem Rückzug auf die Rückzugsebene wird das Werkzeug wieder auf die Bohrungsmitte gefahren und die Spindel in der Bearbeitungsdrehrichtung gestartet.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.4 Dreh– und Bohrzyklen (ISO–Dialekt–T)

3.4.1 Drehzyklen G70 bis G76

Tabelle 3-5 Übersicht der Drehzyklen

G–Befehl	Beschreibung
G70	Schlichtzyklus
G71	Abspanzyklus Längsachse
G72	Abspanzyklus Plansachse
G73	Konturwiederholung
G74	Tieflochbohren und Einstechen in der Längsachse (Z)
G75	Tieflochbohren und Einstechen in der Planachse (X)
G76	Mehrfach–Gewindeschneidzyklus

Die Zyklenversorgung bei G71 bis G76 kann aus zwei G–Befehlen bestehen. Abhängig von den im Satz programmierten Adressen werden nur die Werte der im NC–Satz programmierten Adressen gespeichert, um sie bei einem späteren Zyklenaufruf zu verwenden, oder zusätzlich die Zyklenbearbeitung zu starten. Die Auswertung der beiden Fälle erfolgt im Hüllzyklus.

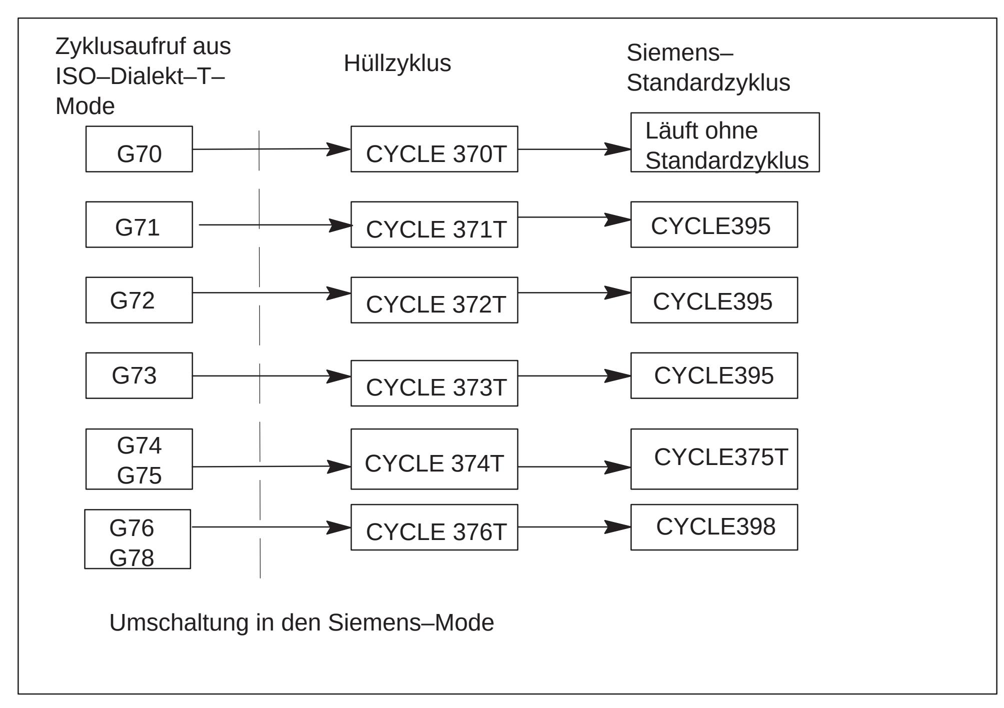


Bild 3-5 Zuordnung des Zyklenaufrufs im ISO–Dialekt-T–Mode über Hüllzyklus zum Siemens Standardzyklus

Schlichtzyklus G70

G70 P.. Q..
- Q..: Satznummer im Hauptprogramm, bei der die Konturbeschreibung endet
- P..: Satznummer im Hauptprogramm, bei der die Konturbeschreibung beginnt

Bild 3-6 Beschreibung der zulässigen Parameter

Im Mode ISO–Dialekt wird bei G70 die Kontur im Hauptprogramm nicht übersprungen. Die Programmweiterbearbeitung erfolgt immer im nächsten Teileprogrammsatz nach dem Zyklenaufruf.

Beispiel:

Mit dem Zyklenaufruf wird die Kontur N20 – N50 abgefahren, nach dem Zyklenende wird das Teileprogramm ab N20 weiterbearbeitet. Sinnvollerweise wird G70 immer nach der Konturbeschreibung aufgerufen.

```
N10 G70 P20 Q50
N20 X100. Z50.
N30 X200.
N40 Z100.
N50 X250. Z111.
N60 M30
```

Die Sätze N20 – N50 werden einmal durch den Schlichtzyklus bearbeitet und anschließend noch einmal durch den normalen Programmablauf ausgeführt.

Abspanzyklus Längsachse G71

G71 U.. R..
- R..: Rückzugsweg
- U..: Zustelltiefe beim Schruppen

Bild 3-7 Beschreibung der zulässigen Parameter; Werte in GUDs speichern

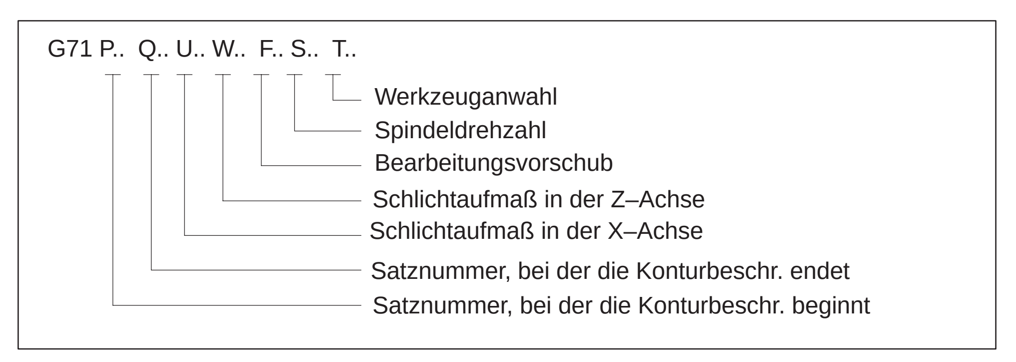


Bild 3-8 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Im Mode ISO–Dialekt–T wird bei G71 die Kontur im Hauptprogramm nach dem Schruppen, mit Schlichtaufmaß abgefahren (Resteckenabspanen). Die Programmweiterbearbeitung erfolgt immer mit dem Teileprogrammsatz nach dem letzten Satz der Konturbeschreibung. Teileprogrammsätze, die zwischen dem Zyklenaufruf und dem ersten Satz der Konturbeschreibung stehen, werden nicht ausgeführt.

```
N10 G71 P50 Q80 U4 W3 F1000 ...
N20 G1 F0.5 G95 S1000
N30 X30. Z10.
N40 M30
N50 X100. Z50.
N60 X200.
N70 Z100.
N80 X250. Z111.
N90 M30
```

Die Sätze N20 – N40 werden übersprungen und nicht ausgeführt. Wird G71 im Teileprogramm nach der Konturbeschreibung aufgerufen, läuft das Programm in einer Endlosschleife. In den Hüllzyklen muß die unterschiedliche Weiterbearbeitung des Teileprogramms berücksichtigt werden.

Beispiel: G71 U6 R5 Werte in GUDs speichern

Im Beispiel werden die programmierten Werte im Hüllzyklus aus den Systemvariablen ($C_xx) gelesen und in kanalspezifiscen GUDs abgelegt. Es gibt für jeden Zyklus (G71–G76) eigene GUDs, damit bleiben die programmierten Werte solange gültig, bis sie das nächstemal (in einem NC–Satz mit G71–G76) programmiert werden. Der Fall 1 muß nicht programmiert werden, deshalb ist es sinnvoll, die zugeordneten GUDs vorzubesetzen.

Beispiel: G71 P30 Q50 U3 Zyklenbearbeitung starten

Im Beispiel werden die programmierten Werte nur in Systemvariablen ($C_xx) abgelegt. Die Systenvariablen werden in jedem NC–Satz, in dem eine G–Funktion für eine Zyklenaufruf programmiert ist überschrieben. Die Zyklenbearbeitung wird bei G71–G73 immer mit dem G–Befehl gestartet, hinter dem die Adressen “P” und “Q” programmiert sind. Die Zyklenbearbeitung G74 – G76 wird mit dem G–Befehl gestartet, hinter dem die Adressen X/U oder Z/W programmiert sind. Die F–, S– und T–Befehle in der Aufrufzeile des Zyklus werden ebenfalls in Systemvariablen abgelegt. Damit müssen die Adressen nicht zyklenspezifisch unterschieden werden. Der Hüllzyklus ordnet den Parametern die Bedeutung zu (z. B. bedeutet bei G76 die Adresse F = Gewindesteigung und nicht Vorschub). Bei G70 sind Vorschub–, Drehzahl– und Werkzeuganwahlbefehle (F, S, T) aus dem Programmteil der Konturbeschreibung relevant.

In beiden Fällen wird immer derselbe Hüllzyklus aufgerufen.

Abspanzyklus Planachse G72

Zulässige Parameter: siehe G71 (Abspanzyklus Längsachse)

G72 W.. R..
- R..: Rückzugsweg
- W..: Zustelltiefe beim Schruppen

Bild 3-9 Beschreibung der zulässigen Parameter; Werte in GUDs speichern

Konturwiederholung G73

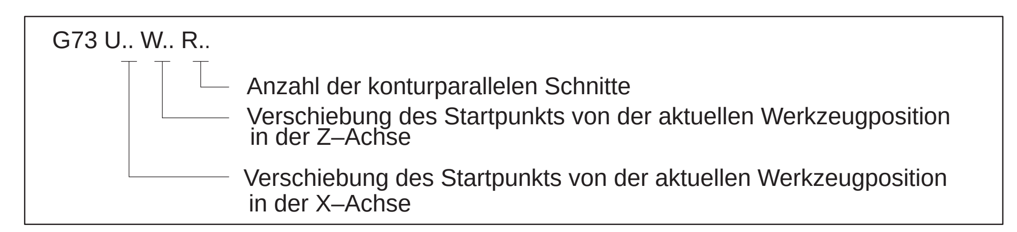


Bild 3-10 Beschreibung der zulässigen parameter; Werte in GUDs speichern

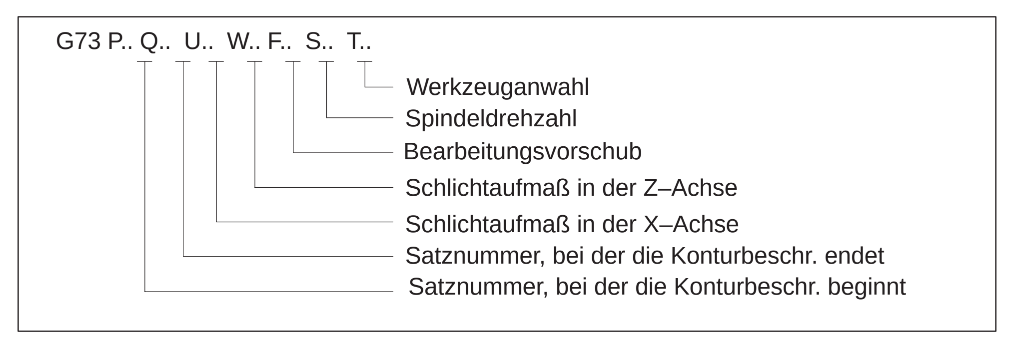


Bild 3-11 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Tieflochbohren und Einstechen in der Längsachse G74

G74 R..
Rückzugsweg zum Späne brechen

Bild 3-12 Beschreibung der zulässigen Parameter; Werte in GUDs speichern

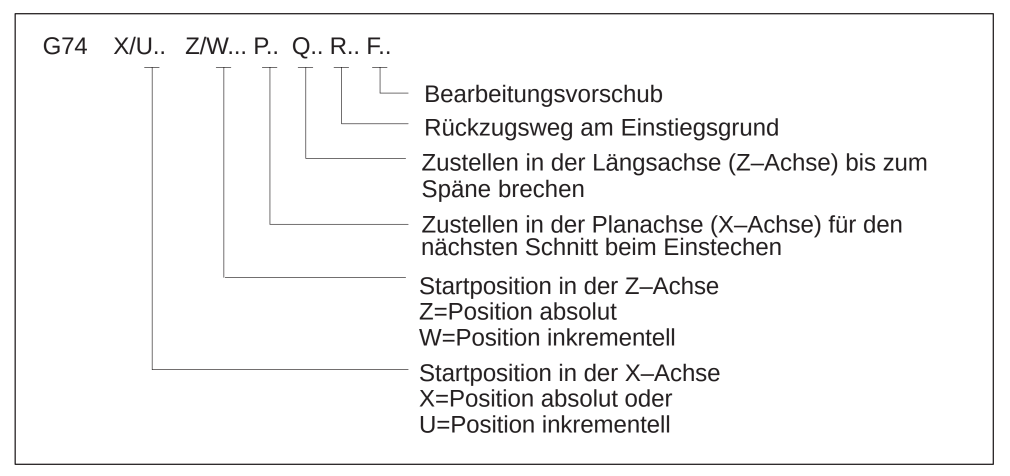


Bild 3-13 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

---

Hinweis

Der Zyklus kann als Bohr– oder Einstechzyklus verwendet werden. Wird der Zyklus zum Bohren eingesetzt, darf die Adresse X/U und P nicht verwendet werden.

Tieflochbohren und Einstechen in der Planachse G75

G75 R..
└── Rückzugsweg zum Späne brechen

Bild 3-14 Beschreibung der zulässigen Parameter; Werte in GUDs speichern

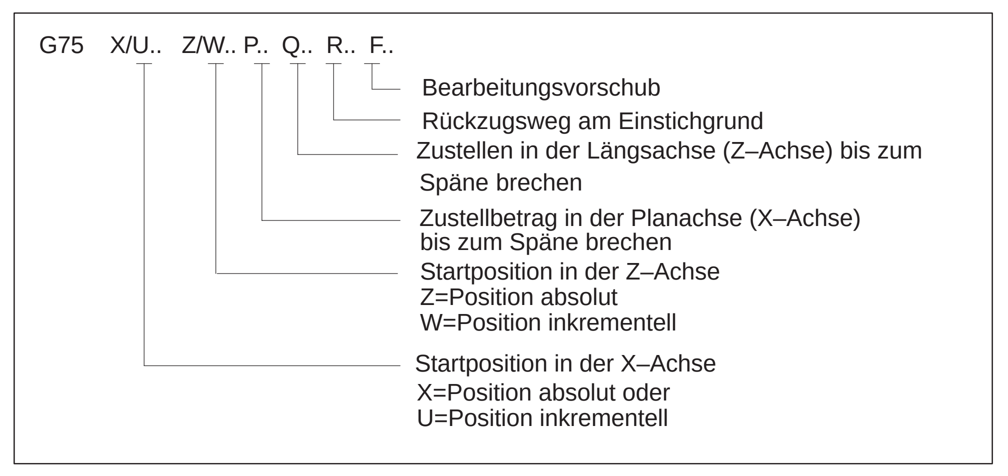


Bild 3-15 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

---

Hinweis

Der Zyklus kann als Bohr– oder Einstechzyklus verwendet werden. Wird der Zyklus zum Bohren eingesetzt, darf die Adresse Z/W und Q nicht verwendet werden.

Mehrfach-Gewindeschneidezyklus G76

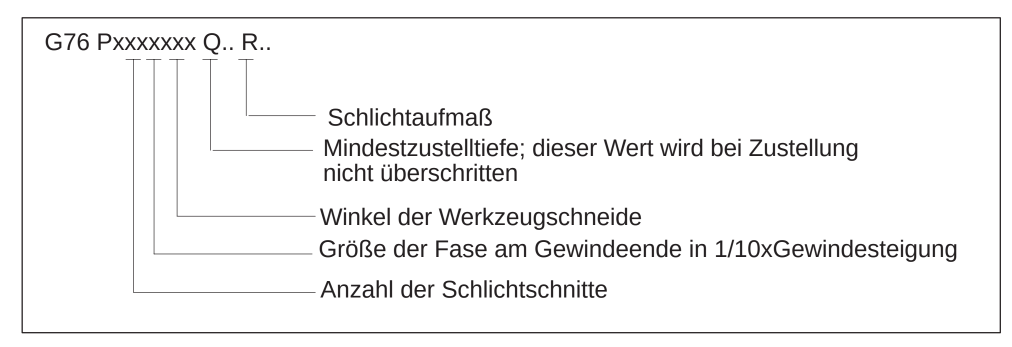


Bild 3-16 Beschreibung der zulässigen Parameter; Werte in GUDs speichern

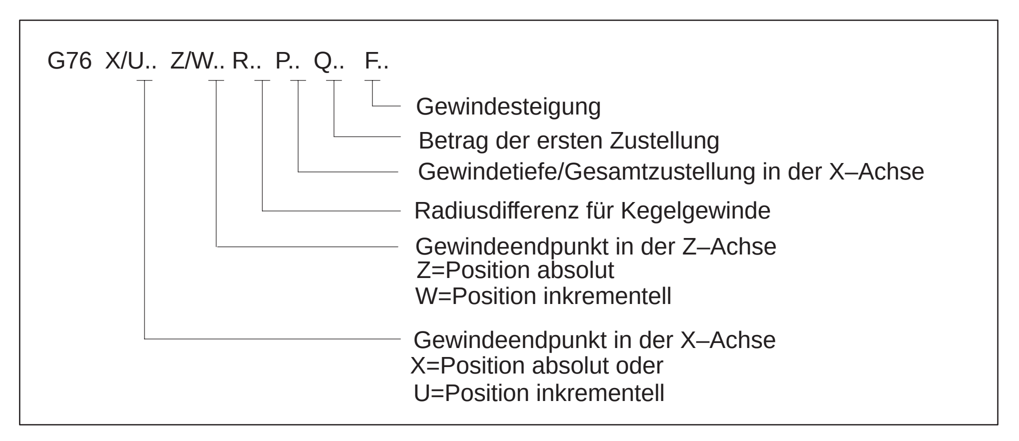


Bild 3-17 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Beispiel für Adresse P:

G76 P012055 Q4 R0.5

P = 012055
Winkel der Werkzeugschneide = 55 Grad
Fase am Gewindeende = 2,0 x Gewindesteigung
1 Schlichtschnitt

3.4.2 Drehzyklen G77 bis G79

Tabelle 3-6 Übersicht der Drehzyklen G77 bis G79

G–Befehl	Beschreibung
G77	Längsabspannen
G78	Gewindeschneiden
G79	Planabspanen

Diese Zyklenaufrufe wirken modal und werden in jedem NC–Satz mit Achsbewegungen aufgerufen. In den Aufrufparametern hinter der G–Funktion werden die Bearbeitungsbewegungen festgelegt. In NC–Sätzen mit Zyklenaufrufen über G77–G79 sind folgende Parameter zulässig:

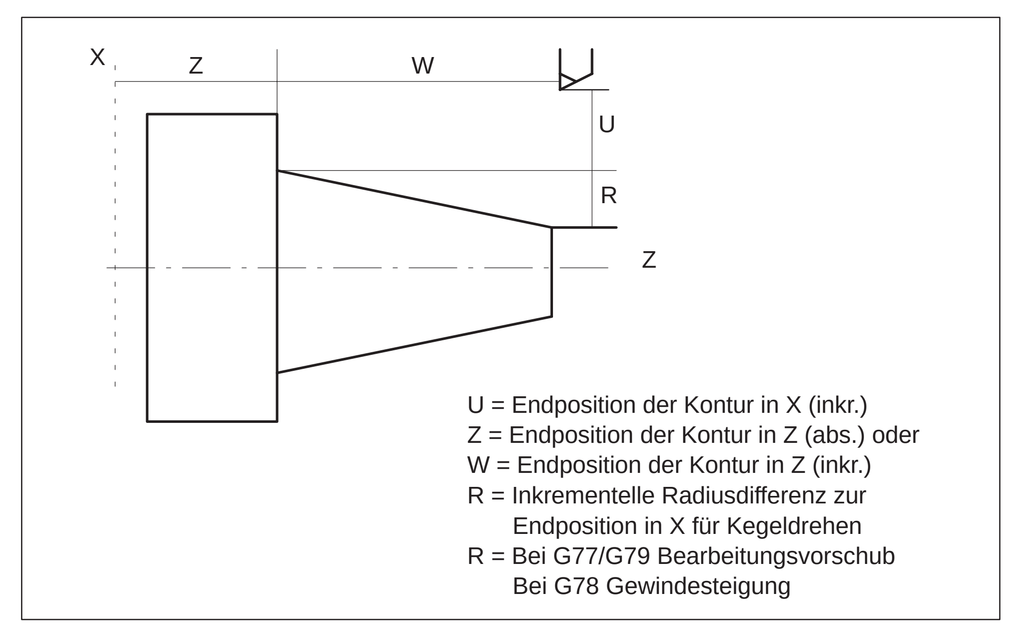


Bild 3-18 Parameter bei Zyklusaufrufen über G77 bis G79

Längsabspanen G77

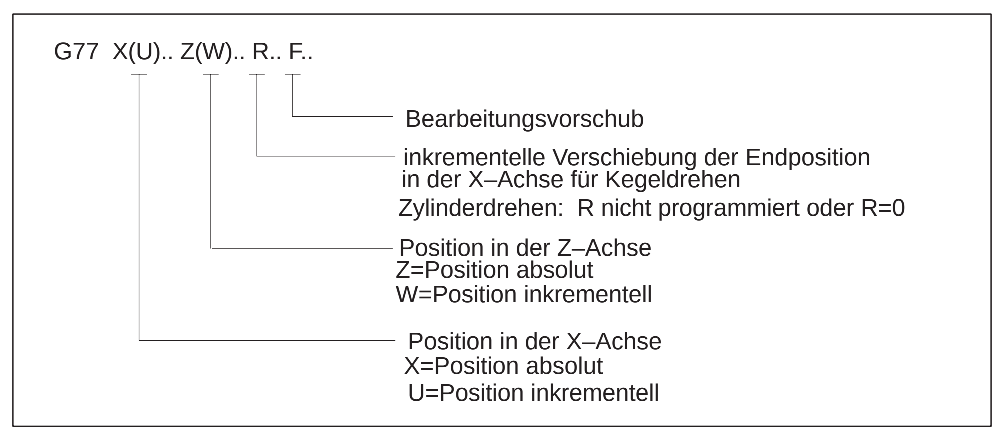


Bild 3-19 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Gewindeschneiden G78

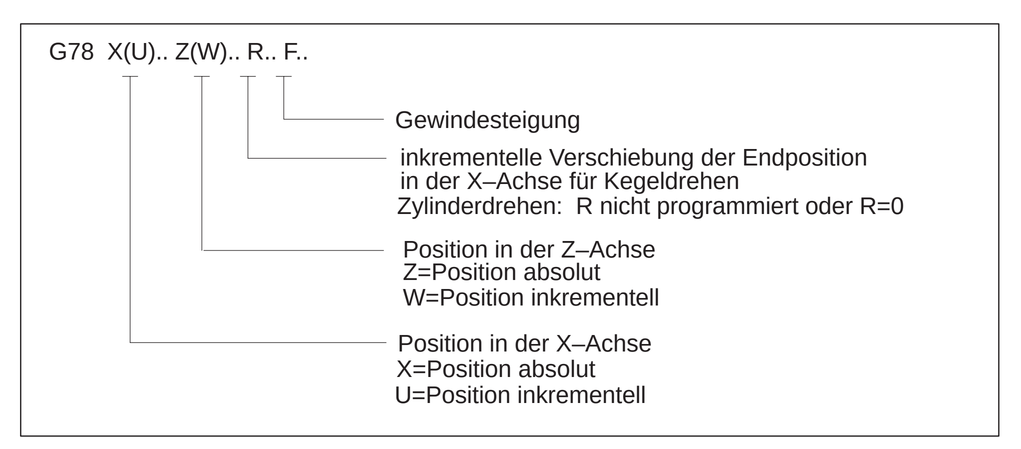


Bild 3-20 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Planabspanen G79

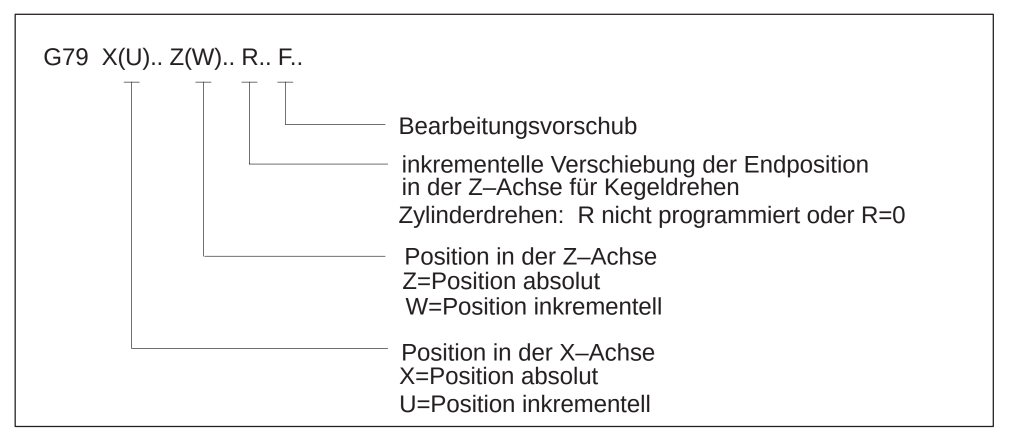


Bild 3-21 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

3.4.3 Bohrzyklen G80 bis G89

Tabelle 3-7 Übersicht der Bohrzyklen

G–Befehl	Beschreibung
G80	Bohrzyklus aus
G83	Stirnflächentieflochbohren
G84	Stirnflächengewindebohren
G85	Stirnflächenbohren
G87	Seitenflächentieflochbohren
G88	Seitenflächengewindebohren
G89	Seitenflächenbohren

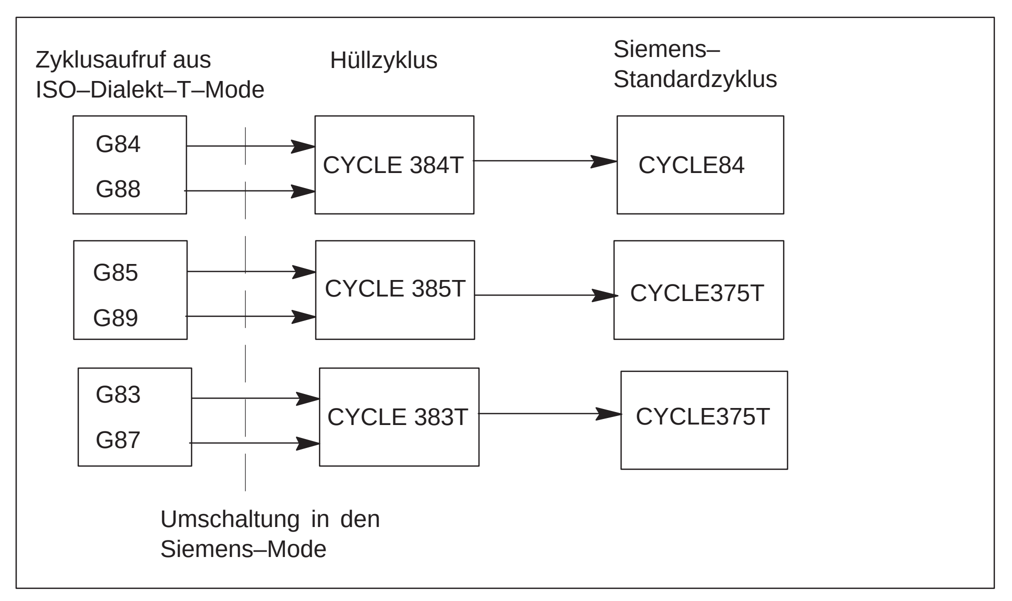


Bild 3-22 Zuordnung der Bohrzyklen im ISO–Dialekt–T–Mode über Hüllzyklus zum Siemens Standardzyklus

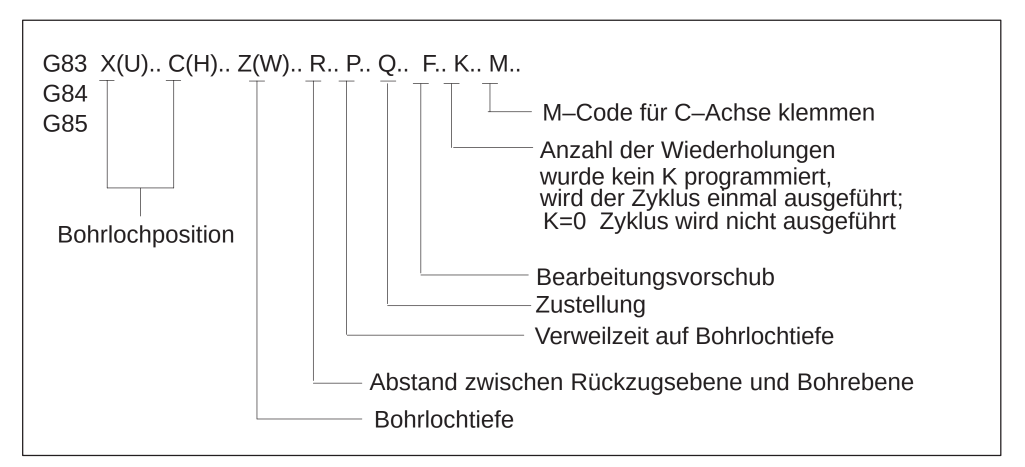


Bild 3-23 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

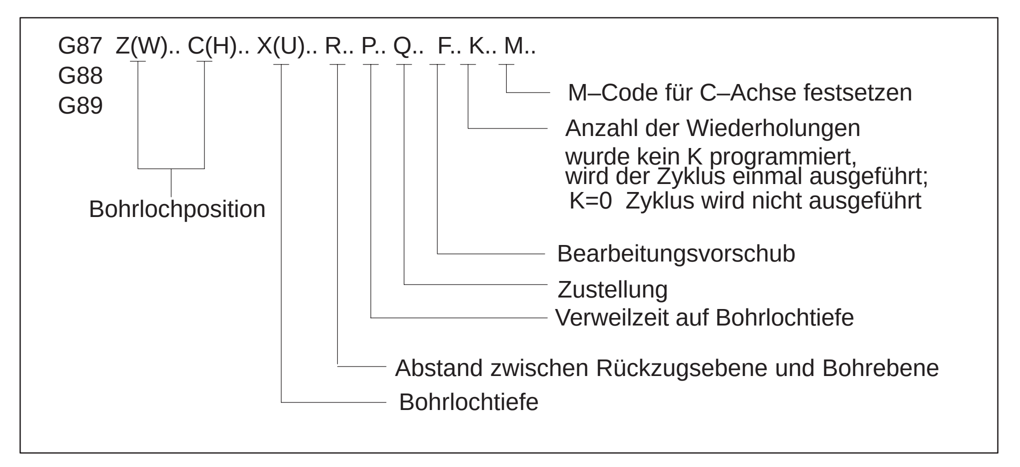


Bild 3-24 Beschreibung der zulässigen Parameter; Zyklenbearbeitung starten

Die Bohrzyklen sind modal wirksam und werden in jedem NC–Satz, in dem Achsbewegungen der Achsen X, C, Z programmiert sind, ausgeführt. Während ein Bohrmodus aktiv ist, müssen bei Parameteränderung nur die neuen Parameter programmiert werden. Die Parameter werden in den Systemvariablen $C_xx (xx = NC–Adresse) abgelegt, die von den Zyklen gelesen werden.

Steht im selben NC–Satz nach der Zyklen–G–Funktion eine G–Funktion der 1. G–Gruppe, wird der Zyklus nicht ausgeführt. Es werden nur die im NC–Satz programmierten Achsen verfahren. Die Adressen R, Q, P, K werden nicht in die Systemvariablen geschrieben. Der in diesem Satz programmierte Vorschub wird aktiviert.

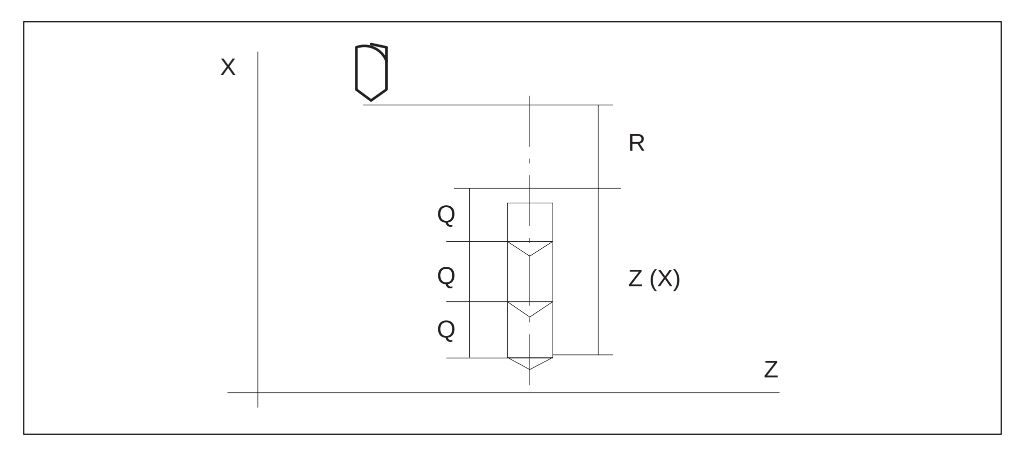


Bild 3-25 Bohrzyklus

Modale Zyklen

Alle modalen Zyklen werden im ISO–Dialekt–Mode mit G80 oder mit einer G–Funktion der 1. G–Gruppe (G00 – G03, G33, G34, außer G77 – G79) abgewählt.

Während ein modaler Zyklus aktiv ist, können in den Folgesätzen Zyklenparameter programmiert werden. Diese Parameter werden in die Systemvariablen geschrieben, damit der Hüllzyklus mit den geänderten Parametern arbeitet.

Beispiel:

```
N10 G81 X10. Z15. R5 Q4 P10 F1000
N20 X50.                  ;Bohrzyklus an der Positon X50
```

3.4.4 Beschreibung Hüllzyklus CYCLE383T

Der Aufruf erfolgt im ISO–Dialekt–T–Mode über die G–Befehle G83 und G87.

Hinweise

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein. Anderenfalls kommt eine Fehlermeldung aus dem Siemens– Bohrzyklus.

Bei ISO–Dialekt–T muß der Programmierer den Sicherheitsabstand beim Definieren der Referenzebene berücksichtigen. Bei Siemens kann der Sicherheitsabstand zur R–Ebene separat angegeben werden.
Diese Möglichkeit wurde auch bei den ISO–Zyklen realisiert. Über GUD_ZSFR[20] kann bei Bedarf ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der R–Ebene berücksichtigt, so muß bei GUD_ZSFR[20] der Wert “NULL” eingegeben werden.

Über GUD_ZSFR[23] wird bestimmt, ob die Verweilzeit (nur bei Tieflochbohren) bei G95 in Sekunden oder in Umdrehungen ausgeführt werden soll.

Ein Hüllzyklus darf nur vom externen G–Code (G83/G87) aufgerufen werden. Ein Aufruf im Siemens–Mode (nach Umschalten mit G290 und Aufruf CYCLE383T) ist unzulässig.

Enthält der G83/G87–Satz andere Achsnamen als X/Z (U/W) kommt der Alarm (61811) “ISO–Achsname nicht zulässig”.

Über GUD_ZSFR[2] wird bestimmt, ob der Zyklus mit Spändebrechen oder Entspänden ausgeführt werden soll.

Bei ISO–Dialekt wird die konstante Einzeltiefe solange beibehalten, bis am Bohrungsgrund der Restbetrag mit einer Zustellung weggenommen wird. Bei Siemens wird der Restbetrag (kleiner 2*Einzeltiefe) am Bohrungsgrund ggf. in zwei gleichgroße Zustellungen aufgeteilt.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.4.5 Beschreibung Hüllzyklus CYCLE384T

Der Aufruf erfolgt im ISO–Dialekt–T–Mode über die G–Befehle G84 und G88.

Hinweise

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein. Anderenfalls kommt eine Fehlermeldung aus dem Siemens– Bohrzyklus. ISO–Dialekt hat hier keine Überwachung.

Bei ISO–Dialekt–T muß der Programmierer den Sicherheitsabstand beim Definieren der Referenzebene berücksichtigen. Bei Siemens kann der Sicherheitsabstand zur R–Ebene separat angegeben werden.
Diese Möglichkeit wurde auch bei den ISO–Zyklen realisiert. Über GUD_ZSFR[20] kann bei Bedarf ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der R–Ebene berücksichtigt, so muß bei GUD_ZSFR[20] der Wert “NULL” eingegeben werden.

Ein Hüllzyklus darf nur vom externen G–Code (G84/G88) aufgerufen werden. Ein Aufruf im Siemens–Mode (nach Umschalten mit G290 und Aufruf CYCLE384T) ist unzulässig.

Enthält der G84/G88–Satz andere Achsnamen als X/Z (U/W) kommt der Alarm (61811) “ISO–Achsname nicht zulässig”.

Über GUD_ZSFI[22] (Wert in %) kann die Bohrdrehzahl beim Rückzug beeinflußt werden.
Beispiel: ZSFI[22]=95 Der Rückzug erfolgt mit 95% von der Bohrtiefe.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.4.6 Beschreibung Hüllzyklus CYCLE385T

Der Aufruf erfolgt im ISO–Dialekt–T–Mode über die G–Befehle G85 und G89.

Hinweise

Die Richtung von Ausgangsebene zur Referenzebene muß mit der Richtung von Referenzebene zur Endtiefe identisch sein. Anderenfalls kommt eine Fehlermeldung aus dem Siemens– Bohrzyklus. ISO–Dialekt hat hier keine Überwachung.

Bei ISO–Dialekt–T muß der Programmierer den Sicherheitsabstand beim Definieren der Referenzebene berücksichtigen. Bei Siemens kann der Sicherheitsabstand zur R–Ebene separat angegeben werden.
Diese Möglichkeit wurde auch bei den ISO–Zyklen realisiert. Über GUD_ZSFR[20] kann bei Bedarf ein Sicherheitsabstand eingegeben werden. Wurde der Sicherheitsabstand bereits bei der Programmierung der R–Ebene berücksichtigt, so muß bei GUD_ZSFR[20] der Wert “NULL” eingegeben werden.

Ein Hüllzyklus darf nur vom externen G–Code (G85/G89) aufgerufen werden. Ein Aufruf im Siemens–Mode (nach Umschalten mit G290 und Aufruf CYCLE385T) ist unzulässig.

Enthält der G85/89–Satz andere Achsnamen als X/Z (U/W) kommt der Alarm (61811) “ISO–Achsname nicht zulässig”.

Enthält der G84–Satz andere Achsnamen als X/Z (U/W) kommt der Alarm (61811) “ISO–Achsname nicht zulässig”.

Hinweis

Alarme sind mit Alarmnummer und Beschreibung im Kap. ”Alarme” aufgelistet.

3.5 Systemvariablen

Die Namen der Systemvariablen beginnen alle mit $C_ xx. Die NC–Adresse deren Wert in der Systemvariablen abgelegt wird, steht in der Namensergänzung xx. In der Variablen $C_G steht immer die G–Nummer mit der ein Zyklus aufgerufen wurde.

Bei allen programmierten Adressen wird in den Systemvariablen $C_x_PROG das Bit0 gesetzt, wenn die Adresse programmiert ist.

Zusätzlich wird bei inkrementell programmierten Achsadressen in der Variablen $C_x_PROG das Bit1 gesetzt.

Bei Unterprogrammende (M17, RET) wird $C_x_PROG auf FALSE gesetzt.

Beispiel 1:

```
N10 G01 G81 X100. Z–50. R20 F100
```

Der Hüllzyklus CYCLE381M für G81 wird automatisch aufgerufen. Im Hüllzyklus werden die Berechnungen durchgeführt und anschließend der Siemens–Standardzyklus CYCLE82 aufgerufen. Der Befehl G01 ist nicht notwendig.

Die Werte der programmierten Adressen werden in die Systemvariablen geschrieben:

Adresse X wird in Systemvariable $C_X geschrieben;
Adresse Z wird in Systemvariable $C_Z geschrieben;
Adresse R wird in Systemvariable $C_R geschrieben;
Adresse F wird in Systemvariable $C_F geschrieben;

Beispiel 2:

Achse Z ist inkrementell programmiert (G91) → $C_Z_PROG=3
Achse Z ist absolut programmiert (G90) → $C_Z_PROG =1

Beispiel 3: Siemenshüllzyklus für Gxy

```
N10 PROC CYCLE377 DISPLOF           ;Satzanzeige bleibt G77–Satz,
                                    ;G–Code–Anzeige einfrieren
N20 DEF REAL DELTA_X, pos_X, pos_Z, VORSCHUB
N30 DEF BOOL R_prog, X_prog, Z_prog
N50 DELTA_X = 0
N60 IF $C_R_PROG                    ;DELTA_X nur dann laden, wenn die
N70  DELTA_X = $C_R                 ;Adresse R programmiert wurde
N75 ENDIF
N110 CYCLE...(DELTA_X, $C_X, $C_Z, $C_R_PROG, $C_X_PROG, $C_Z_PROG,
$C_F)                               ;Siemenszyklus aufrufen
N230 RET                            ;Hüllzyklusende
```

Tabelle 3-8 Liste und Beschreibung der Systemvariablen

Bezeichner	Typ	Beschreibung
$C_A	REAL	Wert der programmierten Adresse A im ISO–Dialekt–Mode für Zyklenprogrammierung
$C_B	REAL	Wert der programmierten Adresse B im ISO–Dialekt–Mode für Zyklenprogrammierung
....	....	.....
$C_G	INT	G–Nummer für Zyklenaufrufe im externen Mode
$C_H	REAL	Wert der programmierten Adresse H im ISO–Dialekt–Mode für Zyklenprogrammierung
$C_I[ ]	REAL	Wert der programmierten Adresse I im ISO–Dialekt–Mode für Zyklenprogrammierung und Makrotechnik mit G65/G66. Für die Makroprogrammierung sind max. 10 Einträge im Satz möglich. Die Werte stehen in der programmierten Reihenfolge im Array.
$C_I_ORDER[ ]	REAL	Beschreibung siehe $C_I[ ], dient zur Festlegung der Programmierreihenfolge
$C_J[ ]	REAL	Beschreibung siehe $C_I[ ]
$C_J_ORDER[ ]	REAL	Beschreibung siehe $C_I[ ], dient zur Festlegung der Programmierreihenfolge
$C_K[ ]	REAL	Beschreibung siehe $C_I[ ]
$C_K_ORDER[ ]	REAL	Beschreibung siehe $C_I[ ], dient zur Festlegung der Programmierreihenfolge
$C_L	REAL	Wert der programmierten Adresse L im ISO–Dialekt–Mode für Zyklenprogrammierung
....	....	....
$C_Z	REAL	Wert der programmierten Adresse Z im ISO–Dialekt–Mode für Zyklenprogrammierung
$C_TS	STRING	String unter der Adresse T programmierten Werkzeugbezeichners
$C_A_PROG	INT	Adresse A ist in einem Satz mit Zyklenaufruf programmiert. 0 = nicht programmiert 1 = programmiert (absolut) 3 = programmiert (inkrementell)
$C_B_PROG	INT	Adresse B ist in einem Satz mit Zyklenaufruf programmiert. 0 = nicht programmiert 1 = programmiert (absolut) 3 = programmiert (inkrementell)
....	....	....
$C_G_PROG	INT	Der Hüllzyklenaufruf ist über eine G–Funktion programmiert
$C_Z_PROG	INT	Adresse Z ist in einem Satz mit Zyklenaufruf programmiert. 0 = nicht programmiert 1 = programmiert (absolut) 3 = programmiert (inkremental)
$C_TS_PROG	INT	Es wurde ein Werkzeugbezeichner unter der Adresse T programmiert TRUE = programmiert, FALSE = nicht programmiert

Tabelle 3-8 Liste und Beschreibung der Systemvariablen

Bezeichner	Typ	Beschreibung
$C_ALL_PROG	INT	Bitmuster aller programmierten Adressen in einem Satz mit Zyklenaufruf Bit 0 = Adresse A Bit 25 = Adresse Z Bit =1 Adresse programmiert Bit = 0 Adresse nicht programmiert
$P_EXTGG[n]	INT	Aktiver G–Code der externen Sprache
$C_INC_PROG	INT	Bitmuster aller inkrementell programmierten Adressen in einem Satz mit Zyklenaufruf Bit 0 = Adresse A Bit 25 = Adresse Z Bit =1 Adresse inkrementell programmiert Bit = 0 Adresse absolut programmiert
$C_I_NUM	INT	Zyklenprogrammierung: Wert ist immer 1, wenn das Bit 0 in $C_I_PROG gesetzt ist. Makroprogrammierung: Anzahl der im Satz programmierten Adresse I (max. 10).
$C_J_NUM	INT	Beschreibung siehe $C_I_NUM
$C_K_NUM	INT	Beschreibung siehe $C_I_NUM
$P_AP	INT	Polarkoordinaten 0=AUS 1=EIN
$C_TYP_PROG	INT	Bitmuster aller programmierten Adressen in einem Satz mit Zyklenaufruf Bit 0 = A Bit 25 = Z Bit = 0 Achse ist als INT programmiert Bit = 1 Achse ist als REAL programmiert
$C_PI	INT	Programmnummer der Interruptroutine, die mit M96 programmiert wurde

3.6 Programmierung von Konturzügen (ISO–Dialekt–T)

Konturzüge können sowohl im ISO–Dialekt–T–Mode als auch im Siemens–Mode programmiert werden.

Es gibt 3 Grundformen der Konturzüge

- eine Gerade
  Endpunkt wird mit einer kartesichen Koordinate und einem Winkel programmiert
- zwei Geraden
  Übergang mit einer Rundung oder eine Fase
- drei Geraden
  Übergänge mit einer Rundung oder einer Fase,

In den folgenden Beschreibungen werden die Adressbuchstaben X, Z, A, R und C teilweise mit Indizes versehen, um eine eindeutige Zuordnung zwischen NC–Satz und der zugehörigen Zeichnung zu ermöglichen. Im NC–Programm erscheinen diese Indizes nicht. Die Zuordnung ergibt sich immer eindeutig aus dem Satz, in dem ein Adressbuchstabe enthalten ist. Der Adressbuchstabe Q steht als Platzhalter an den Stellen, an denen wahlweise R oder C stehen kann. Q kann auch entfallen. Am Übergang der beiden beteiligten Geraden wird dann keine Fase bzw. Rundung eingefügt.

In den Sätzen, die Konturzüge beschreiben, können beliebige weitere NC–Adressen verwendet werden wie z.B. Adressbuchstaben für weitere Achsen (Einzelachsen oder Achse senkrecht zur Bearbeitungsebene), Hilfsfunktionsangaben, G–Codes, Geschwindigkeiten usw.

In den folgenden Beispielen wird davon ausgegangen, daß G18 aktiv ist. Die Programmierung von Konturzügen ist jedoch ohne Einschränkungen auch bei G17 oder G19 möglich.

ISO–Dialekt–Mode

Die Adresse C dient im ISO–Dialekt–Mode sowohl als Achsbezeichner als auch als Bezeichner für eine Fase im Konturzug.

Die Adresse R kann ein Zyklenparameter sein oder ein Bezeichner für den Radius im Konturzug.

Zur Unterscheidung der beiden Möglichkeiten muß bei der Konturzugprogrammierung (wie bei ISO–Dialekt) ein “,” vor die Adresse C oder R gesetzt werden. Ist ein Winkel vor C oder R programmiert, muß kein Komma davor programmiert werden. Werden Radius und Fase zusammen in einem Satz programmiert, z. B. N333 X100 A10 C20 R15 wird unabhängig von der Programmierreihenfolge, immer ein Radius in die Kontur eingefügt. Die Fase wird ignoriert.

Siemens–Mode

Im Siemens–Mode werden die Bezeichner für Winkel, Radius und Fase über Maschinendaten definiert. Dadurch werden Namenskonflikte vermieden. Es darf kein Komma vor dem Bezeichner für Radius oder Fase programmiert werden.

Hinweis

MD 10652 für Winkel: $MN_CONTEUR_DEF_ANGLE_NAME
MD 10654 für Radius: $MN_RADIUS_NAME
MD 10656 für Fase: $MN_CHAMFER_NAME
(gilt nur im Siemens Mode)

3.6.1 Endpunktprogrammierung mit Winkeln

Erscheint in einem NC–Satz der Adressbuchstabe A, so dürfen zusätzlich keine, eine oder beide Achsen der aktiven Ebene programmiert sein.

Ist keine Achse der aktiven Ebene programmiert, so handelt es sich entweder um den ersten oder zweiten Satz eines Konturzuges, der aus zwei Sätzen besteht. Handelt es sich um den zweiten Satz eines solchen Konturzuges, so sind Start– und Endpunkt in der aktiven Ebene identisch. Der Konturzug besteht dann nur aus einer Bewegung senkrecht zur aktiven Ebene.

Ist genau eine Achse der aktiven Ebene programmiert, so handelt es sich entweder um eine einzelne Gerade, deren Endpunkt eindeutig aus dem Winkel und der programmierten kartesischen Koordinate bestimmt ist, oder um den zweiten Satz eines aus zwei Sätzen bestehenden Konturzuges. Im zweiten Fall wird die fehlende Koordinate gleich der letzten erreichten (modalen) Position gesetzt.

Sind zwei Achsen der aktiven Ebene programmiert, handelt es sich um den zweiten Satz eines Konturzuges, der aus zwei Sätzen besteht. Ging dem aktuellen Satz kein Satz mit Winkelprogrammierung ohne programmierte Achsen der aktiven Ebene voraus, so ist ein solcher Satz nicht zulässig.

Der Winkel A darf nur bei Linear– oder Splineinterpolation programmiert werden. (Splineinterpolation nur im Siemens–Mode.)

In folgenden Fällen werden Alarme generiert:

- Bei einem aus zwei Sätzen bestehenden Konturzug wird beim Übergang vom ersten zum zweiten Satz die aktive Ebene gewechselt.
- Mit den programmierten Winkeln kann bei einem aus zwei Geraden bestehenden Konturzug kein gültiger Zwischenpunkt gebildet werden.
- In einem Satz mit der Adresse A ist weder Linear– noch Splineinterpolation aktiv.
- Auf einen Satz mit der Adresse A ohne programmierte Achse in der aktiven Ebene folgt kein Satz, mit dem der Endpunkt des Konturzuges bestimmt wer-

den kann. Das ist dann der Fall, wenn dieser Satz der letzte eines Programms ist, oder wenn der Folgesatz einen Vorlaufstop enthält.

- Im zweiten Satz eines aus zwei Geraden bestehenden Konturzuges wurde kein Winkel programmiert.
- In einem Satz mit der Adresse A, der nicht der zweite Satz eines aus zwei Geraden bestehenden Konturzuges ist, sind beide Achsen der aktiven Ebene programmiert.
- Programmierte kartesische Koordinate und programmmierter Winkel sind inkompatibel.

3.6.2 Gerade mit Winkel

Der Endpunkt wird definiert durch Angabe des Winkels A und einer der beiden Koordinaten X_2 oder Z_2.

Programmiersyntax:

X_2.. A.. oder
Z_2.. A..

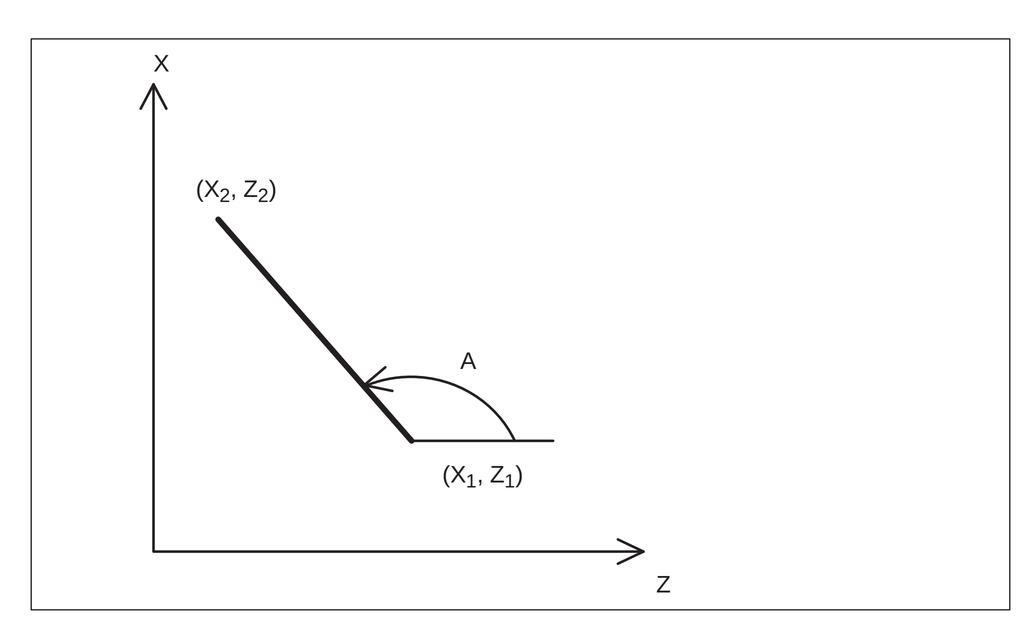


Bild 3-26 Gerade mit Winkel

Beispiel (Bild 3-27):

Programmierung im ISO–Dialekt–T–Mode:

```
N10 G1 X5. Z70. F1000 G18
N 20 X88.8 A 110 oder (Z39.5 A110)
```

Programmierung im Siemensmode:

```
N10 X5. Z70. F1000 G18
```



```
N20 X88.8 ANG=110 oder (Z39.5 ANG=110)
```

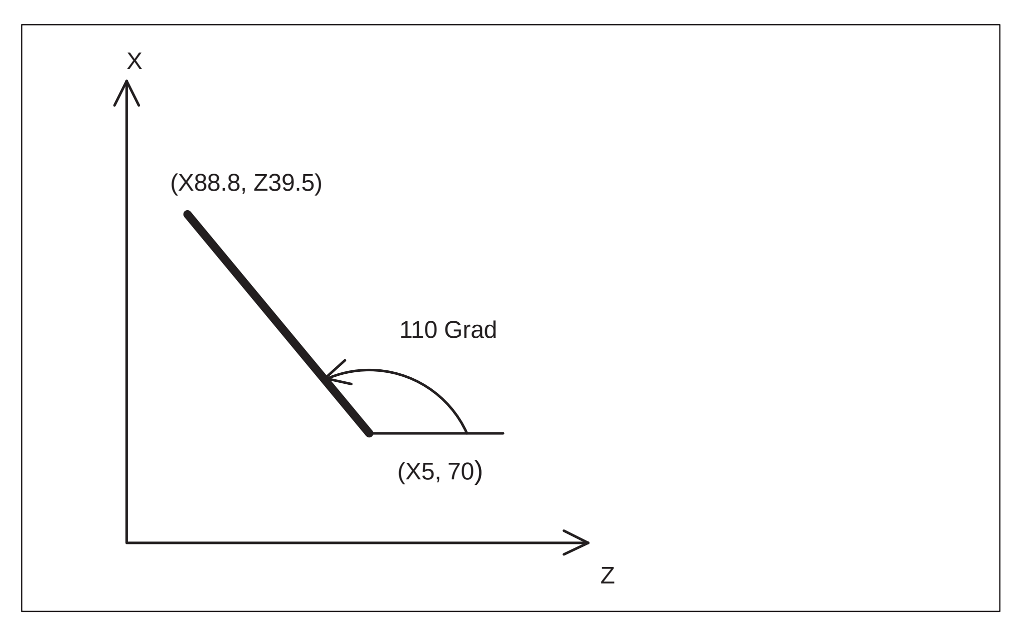


Bild 3-27 Gerade mit Winkel

3.6.3 Zwei Geraden

Der Endpunkt der ersten Geraden kann durch Angabe der kartesischen Koordinaten oder durch Angabe der Winkel der beiden Geraden relativ zur Abszisse programmiert werden.

Programmiersyntax:

```
N10  A1..   (Q..)
N20  X3..  Z3..  A2..
```

bzw.

```
N10  X1.. Z1..  (Q..)
N20  X3..  Z3..
```

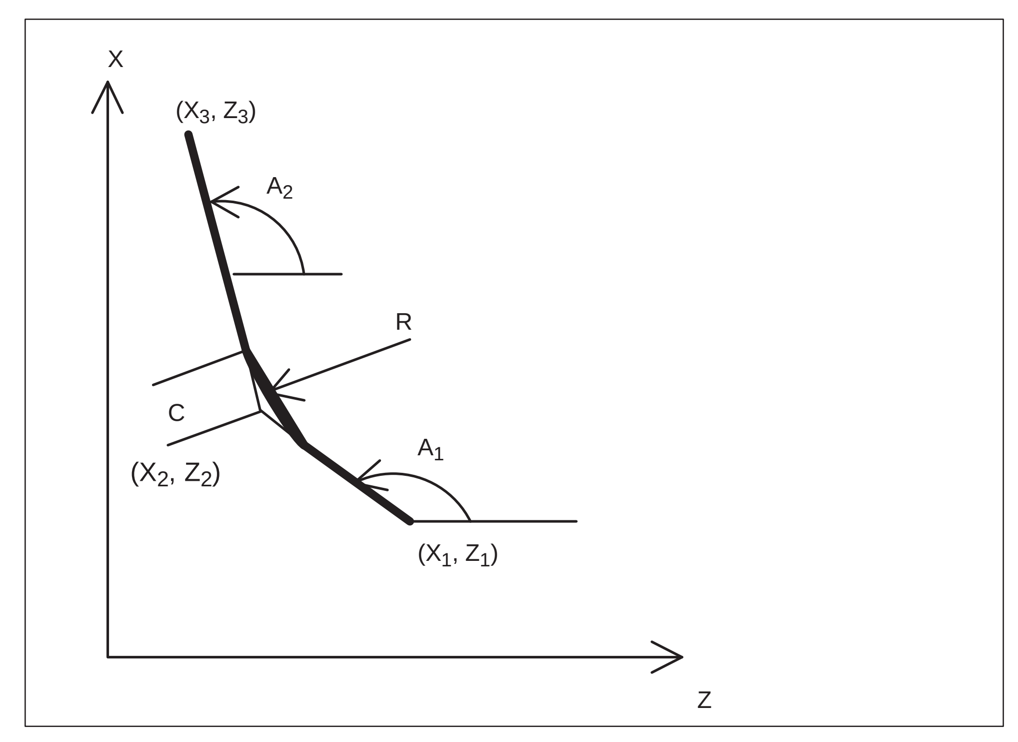


Bild 3-28 Zwei Geraden

Beispiel (Bild 3-29):

Programmierung im ISO–DIALEKT–T–Mode:

```
N10 G1 X10. Z80. F1000 G18
N20 A 1.48.64 C5.5
N30 X85. Z40. A100
```

Programmierung im Siemensmode:

```
N10 X10. Z80. F1000 G18
N20 ANG=148.65 CHR=5.5
N30 X85. Z40. ANG=100
```

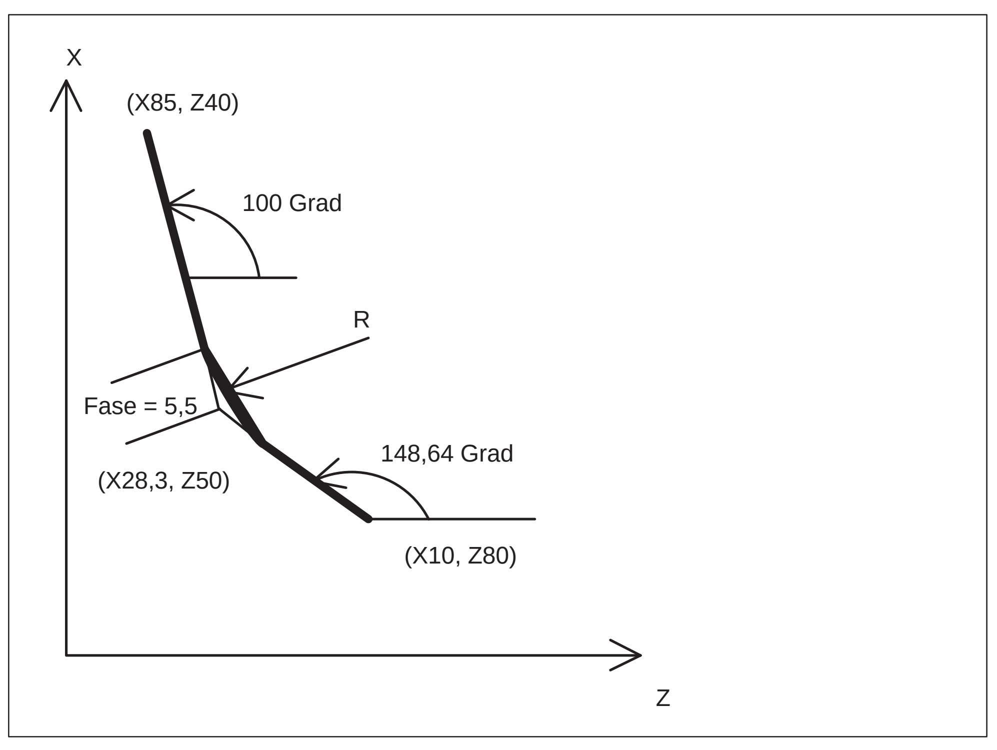


Bild 3-29 Zwei Geraden

3.6.4 Drei Geraden

Der Endpunkt der dritten Gerade muß kartesisch programmiert werden. Der Übergang von der zweiten zur dritten Koordinate kann wahlweise wieder mit einer Fase oder einem Radius erfolgen.

Diese Art der Programmierung kann für beliebig viele weitere Sätze fortgesetzt werden, d.h es muß nicht zwischen Konturzügen mit zwei oder mehr Sätzen unterschieden werden.

Programmiersyntax:

N10 X_2.. Z_2.. (Q_1..)
N20 X_3.. Z_3.. (Q_2..)
N30 X_4.. Z_4..

bzw.

N10 A_1.. (Q_1..)
N20 X_3.. Z_3.. A_2.. (Q_2..)
N30 X_4.. Z_4..

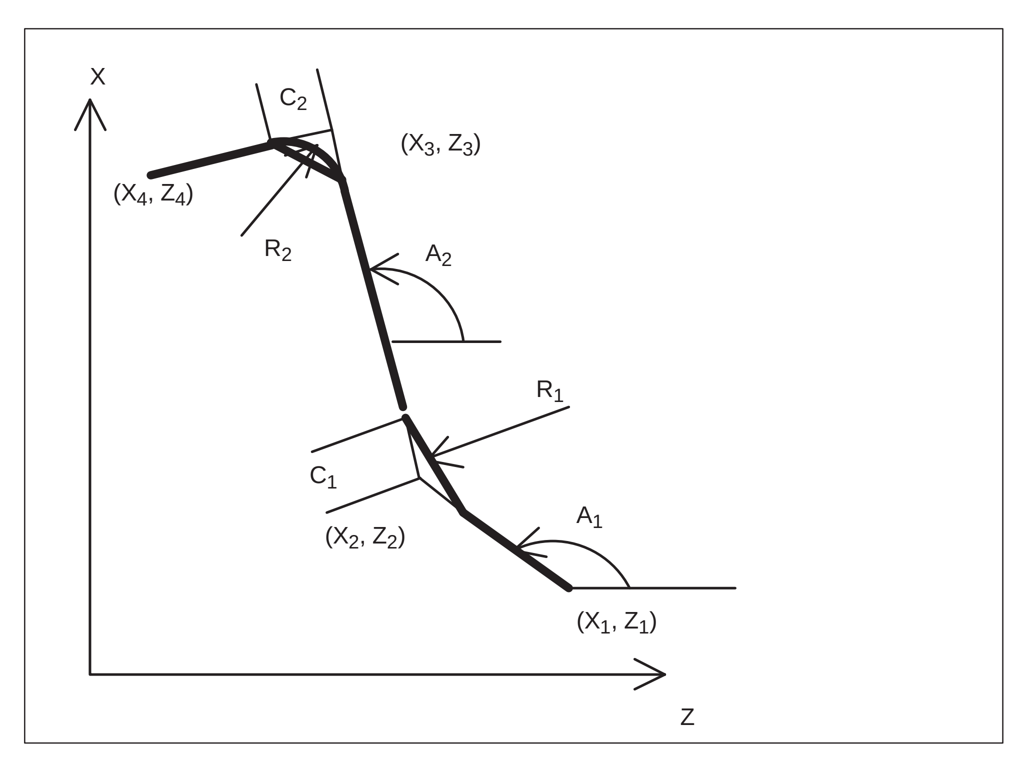


Bild 3-30 Drei Geraden

Beispiel (Bild 3-31):

Programmierung im ISO–DIALEKT–T–Mode:

```
N10 G1 X10. Z100. F1000 G18
N20 A140 C7.5
N30 X80. Z70. A95.824, R10
```

Programmierung im Siemensmode:

```
N10 X10. Z100. F1000 G18
N20 ANG=140 CHR=7,5
N30 X80. Z70. ANG=95.824 RND=10
```

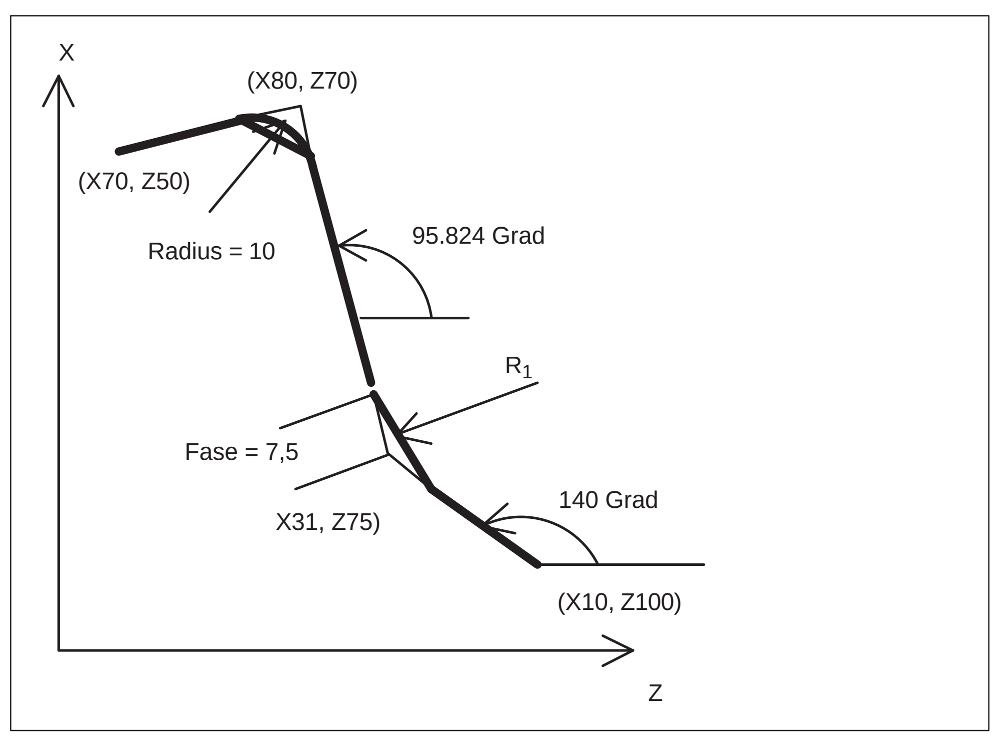


Bild 3-31 Drei Geraden

3.6.5 Polygondrehen mit G51.2

Mit der Funktion G51.2 können durch die Kopplung von zwei Spindeln mehrkantige Werkstücke hergestellt werden. Dies entspricht der Synchronspindel–Funktion im Siemensmode mit einem Übersetzungsverhältnis ungleich 1 : 1.

Mit der Programmiersyntax G51.2 Q.. P.. R.. wird die Synchronspindelkopplung eingeschaltet. Das Übersetzungsverhältnis von Leitspindel zu Folgespindel wird mit den Parameter ”Q” und ”P” festgelegt. Soll die Kopplung mit einem Winkelversatz von Folgespindel und Leitachse eingeschaltet werden, wird die Winkeldifferenz mit der Adresse ”R” programmiert.

Bei der Abbildung der Funktion auf die Siemenssprache müssen zum Aktivieren der Synchronspindelfunktion immer zwei Teileprogrammbefehle abgesetzt werden, die nicht im selben Satz stehen dürfen.
Mit einem Teileprogrammbefehl wird die Zuordnung von Leit- und Folgespindel und das Übersetzungsverhältnis (und Kopplungsart) definiert (COUPDEF(..)). Der zweite Teileprogrammbefehl schaltet die Kopplung mit dem programmierten Winkelversatz ein (COUPON(..)). Um die beiden Programmbefehle auszuführen wird mit G51.2 ein Zyklus aufgerufen (CYCLE3512). Die programmierten Werte werden in den Zyklenparametern $C_P, $C_Q und $C_R übergeben. Mit G50.2 wird die Kopplung wieder ausgeschaltet (ebenfalls mit CYCLE3512).

Mit der Programmierung von G51.2 wird immer die 1. Spindel im Kanal als Leitspindel und die 2. Spindel als Folgespindel definiert. Als Kopplungsart wird Sollwertkopplung angewählt.

Beispiel

```
N10 T1234
N20 G0 X10 Z100 M3 S1000
N30 G51.2 P1 Q3     ; Synchronspindel mit Übersetzungsverh. 1 : 3 und Winkelversatz 0
                    : Grad starten
Nxx ....
N1000 G51.2 R180    ; Winkelversatz zwischen Leit– und Folgespindel von
                    ; 180 Grad

N1200 G50.2         ;Synchronspindelbetrieb ausschalten
N2000 M30
```

Ausführliche Beschreibung der Synchronspindel–Funktion siehe

/FB2/ SINUMERIK 840D/810D(CCU2)
Funktionsbeschreibung Erweiterungsfunktionen, Kapitel M1 und
/PGA/ SINUMERIK 840D/810D
Programmieranleitung Arbeitsvorbereitung, Kapitel ”Synchronspindel”

3.6.6 Konturwiederholung G72.1 / G72.2

Mit G72.x wird ein unter der Adresse P.. programmiertes Unterprogramm aufgerufen. Mit der Adresse L.. wird die Anzahl der Unterprogrammwiederholungen festgelegt. Wird die Adresse L nicht programmiert, wird das Unterprogramm einmal ausgeführt. Vor jedem Unterprogrammaufruf wird abhängig vom G–Code eine Koordinatendrehung ausgeführt (G72.1) oder ein inkrementeller Weg, bezogen auf den Startpunkt der Kontur, verfahren (G72.2).

G72.1

Mit G72.1 wird ein Unterprogramm (in dem die zu wiederholende Kontur programmiert ist) mehrfach aufgerufen. Vor jedem Unterprogrammaufruf wird das Koordinatensystem um einen bestimmten Winkel gedreht.

Diese Funktion wird durch den Aufruf eines Zyklus realisiert (CYCLE3721). Dem Zyklus werden die programmierten Werte in den Zyklenparametern $C_.. übergeben. Die G-Funktionsnummer steht in $C_G, dabei wird für G72.1 der Wert 721 und für G72.2 der Wert 722 in $C_G eingetragen. Der Zyklus führt die Koordinatendrehung n-mal aus und ruft das Unterprogramm n-mal auf. Die Koordinatendrehung wird um die auf der angewählten Ebene senkrecht stehenden Achse ausgeführt.

X.. Y.. (Z..) Bezugspunkt für Koordinatendrehung
P.. Unterprogrammnummer

L.. Anzahl der Unterprogrammwiederholungen
R.. Drehwinkel

Beispiel:

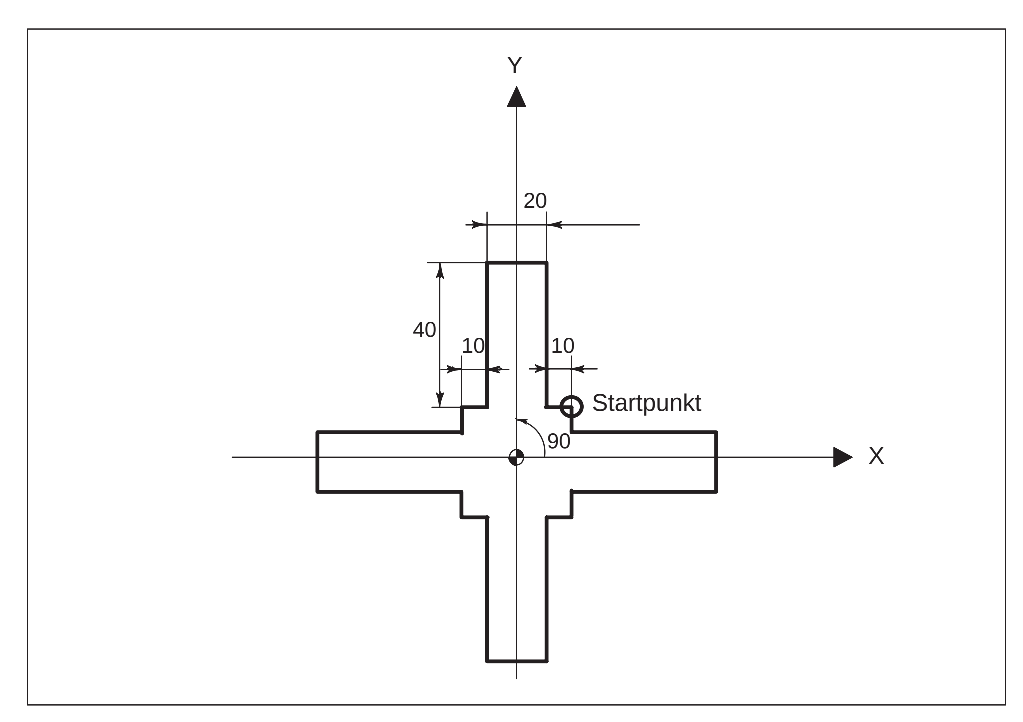


Bild 3-32 Konturwiederholung mit G72.1

Hauptprogramm

```
N10 G92 X40.0 Y50.0 ;
N20 G01 G90 G17 G41 20 Y20 D01 F1000
N30 G72.1 P1234 L4 X0 Y0 R90.0
N40 G40 G01 X100 Y50 Z0
N50 G00 X40.0 Y50.0 ;
N60 M30 ;
```

Unterprogramm 1234.spf

```
N100 G01 X10
N200 Y50
N300 X–10
N400 Y10
N500 X–20
N600 M99
```

G72.2

Mit G72.2 wird ein Unterprogramm (in dem die zu wiederholende Kontur programmiert ist) mehrfach aufgerufen. Vor jedem Unterprogrammaufruf werden die mit I, J, K programmierten Achsen inkrementell verfahren. Im CYCLE3721 wird durch die programmierte G-Funktion erkannt, ob das Konturunterprogramm nach einer Rotation oder einer Linearverschiebung wiederholt wird.

Diese Funktion wird durch den Aufruf eines Zyklus realisiert. Dem Zyklus werden die programmierten Werte in den Zyklenparametern $C_.. übergeben. Der Zyklus ruft das Unterprogramm n-mal auf. Vor jedem Unterprogrammaufruf wird ein unter I, J, K programmierter Weg inkrementell vom Startpunkt aus gerechnet, verfahren.

I.. J.. K..	Position, auf die die Achsen X, Y, Z vor Aufruf des Unterprogramms verfahren werden. Die Position bezieht sich auf den Startpunkt des Unterprogramms.
P..	Unterprogrammnummer
L..	Anzahl der Unterprogrammwiederholungen

Beispiel:

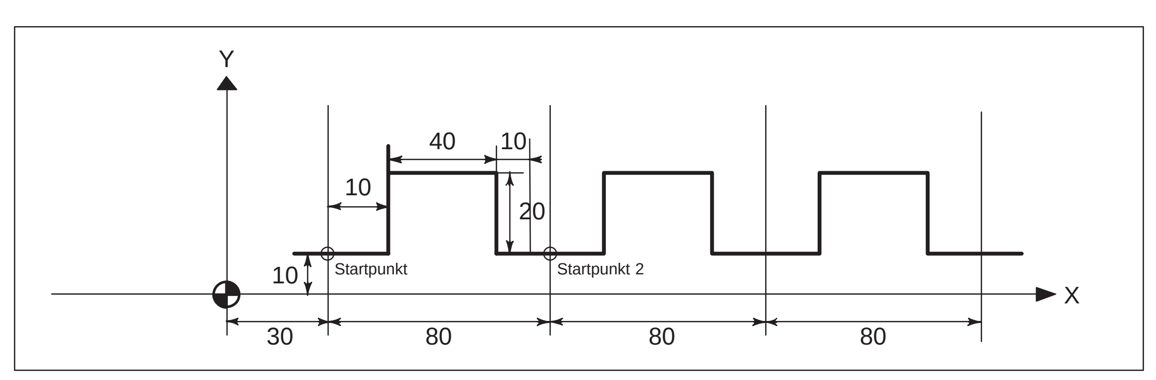


Bild 3-33 Konturwiederholung mit G72.2

```
N10 G00 G90 X0 Y0
N20 G01 G17 G41 X30. Y0 D01 F1000
N30 Y10.
N40 X30.
N50 G72.2 P2000 L3 I80. J0

O2000 G90 G01 X40.
N100 Y30.
N200 G01 X80.
N300 G01 Y10.
N400 X90.
N500 M99
```

■

4 Inbetriebnahme

4.1 Maschinendaten

ISO–Dialekt–Mode aktivieren

MD 18800: $MN_MM_EXTERN_LANGUAGE

Hinweis

Die Umschaltung auf eine externe Programmiersprache ist eine Option.

Auswahl von ISO–Dialekt–M oder T

MD 10880: $MN_MM_EXTERN_CNC_SYSTEM=1 ISO–Dialekt–M
MD 10880: $MN_MM_EXTERN_CNC_SYSTEM=2 ISO–Dialekt–T

Achsnamen ISO–Dialekt–T

Die maximale Achsanzahl beträgt 8. Die Achsbezeichner der ersten 2 Achsen sind fest X, Z. Mögliche Achsbezeichner für die weiteren Achsen sind Y, A, B, C, U, V, W.

Ist G Codesystem A aktiv (hierbei gibt es kein G91) werden inkrementelle Werte für X, Z, Y mit U, V, W programmiert. U, V, W können dann nicht als Achsbezeichner verwendet werden, es sind maximal 6 Achsen möglich. Mit H wird die C–Achse inkrementell verfahren.

Ist B kein verwendeter Achsbezeichner, kann B als erweiterte Hilfsfunktion benutzt werden. B wird dann als Hilfsfunktion H mit der Adresserweiterung 1 (H1=) ausgegeben.

Konturzug:
Mit den Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK, Bit0 wird die Progammierung des Konturzuges festgelegt.

0: Konturzug wird mit ,C ,R ,A programmiert (mit Komma). C und A können Achsbezeichner sein.
1: Konturzug wird mit C R A programmiert ohne Komma). C und A können keine Achsbezeichner sein.

Achsnamen ISO–Dialekt–M

Die maximale Achsanzahl beträgt 8. Die Achsbezeichner der ersten 3 Achsen sind fest X, Y, Z. Mögliche Achsnamen für die weiteren Achsen sind A, B, C, U, V, W.

Ist B keine Hilfsfunktion kann B als Achsbezeichner verwendet werden.

Achsinterpolation

Standardmäßig interpolieren bei ISO–Dialekt alle programierbaren Achsen miteinander.

Das entspricht FGROUP bei ISO–Dialekt–M: X, Y, Z (A, B, C, U, V, W).
Das entspricht FGROUP bei ISO–Dialekt–T: X, Z, Y (C).

Mit dem Maschinendatum 4002 $MC_FGROUP_DEFAULT_AXES[0] wird dieses Verhalten erreicht, wenn die Maschine 4 Achsen hat:

$MC_FGROUP_DEFAULT_AXES[0] = 1
$MC_FGROUP_DEFAULT_AXES[1] = 2
$MC_FGROUP_DEFAULT_AXES[2] = 3
$MC_FGROUP_DEFAULT_AXES[3] = 4

Nullpunktverschiebung (nur ISO–Dialekt–M)

Sollen nur die NVs G54 bis G59 benutzt werden, muß MD
28080: $MC_MM_NUM_USER_FRAMES bzw. bei globalen Frames
18601: $MC_MM_NUM_GLOBAL_USER_FRAMES >= 7 gesetzt werden.

Soll nach Reset G54 aktiv sein müssen die folgenden MD auf 1 gesetzt werden:
20154: $MC_EXTERN_GCODE_RESET_VALUES[13]
20150: $MC_GCODE_RESET_VALUES[7]

Werden die erweiterten Nullpunktverschiebungen G54 P1 bis P48 benutzt, muß
28080: $MC_MM_NUM_USER_FRAMES bzw. bei globalen Frames
18601: $MC_MM_NUM_GLOBAL_USER_FRAMES >= 55 gesetzt werden.

Werden die nachfolgenden MDs auf 7 gesetzt
20154: $MC_EXTERN_GCODE_RESET_VALUES[13]
20150: $MC_GCODE_RESET_VALUES[7]
ist nach Reset G54 P1 aktiv. G54 P1 wird auf Siemens G507 abgebildet.

Soll eine angewählte Nullpunktverschiebung nicht mit G91 herausgefahren werden, ist das folgende Settingdatum auf 0 zu setzen
42440: $SC_FRAME_OFFSET_INCR_PROG

Werkzeuglängen und Radiuskorrektur mit G53 unterdrücken:
10760: $MN_G53_TOOLCORR = 1

Umschaltung Metrisch/Inch

Die Handradbewertung und Inkrementbewertung werden nicht mit G20 und G21 umgeschaltet. Diese Umschaltung muß von der PLC aus durchgeführt werden: siehe MD 31090: $MA_JOG_INCR_WEIGHT.

Im ISO–Dialekt–Mode werden die Nullpunktverschiebungen bei der Umschaltung umgerechnet. Bei ISO–Dialekt–Original erfolgt lediglich eine Kommaverschiebung.

Durchmesser oder Radius programmieren

20150: Mit MD $MC_GCODE_RESET_VALUES[28] = 2
wird die Durchmesser–Programmierung für die Planachse aktiviert.

Dezimalpunktprogrammierung

Mit dem Maschinendatum 10884: $MN_EXTERN_FLOATINGPOINT_PROG wird zwischen Standard Notation und PocketcalculatorNotation gewählt.

Die Eingabefeinheit IS–B und IS–C wird mit dem Maschinendatum 10886: $MN_EXTERN_INCREMENT_SYSTEM ausgewählt.

Zu beachten ist, daß die MD 10200 $MN_INT_INCR_PER_MM und 10210: $MN_INT_INCR_PER_DEG mindestens so eingestellt sind, daß die Feinheit IS–B oder IS–C verrechnet werden kann.
Ansonsten werden die programmierten Werte gerundet.

Beispiel: IS–C mm $MN_INT_INCR_PER_MM = 10000.

Skalierung

22910: $MC_WEIGHTING_FACTOR_FOR_SCALE=0 0,001
22910: $MC_WEIGHTING_FACTOR_FOR_SCALE=1 0,00001

Achsialer Skalierungsfaktor: 43120: $MA_DEFAULT_SCALE_FACTOR_AXIS

Skalierungsfaktor P: 42140: $MC_DEFAULT_SCALE_FACTOR_P

Achsiales Skalieren freigeben: 22914: $MC_AXES_SCALE_ENABLE = 1 (bei = 0 achsiales Skalieren nicht möglich)

Position im Maschinenkoordinatensystem G53

Die Achsgeschwindigkeit zum Positionieren bei G53 und bei G00 ohne Interpolation wird im MD 32060: $MA_POS_AX_VELO festgelegt.

Mit G53 X.. Y... wird eine Position im Maschinenkoordinatensystem angefahren. Die Achsen interpolieren nicht miteinander, sondern jede Achse fährt getrennt mit maximaler Geschwindigkeit auf die programmierte Position.

Inkrementelle Positionen werden im G53 Satz überlesen. Während die Werkzeugradiuskorrektur bzw. Längenkorrektur aktiv ist (G41/G42, G43/G44) werden die Achsen **nicht** unabhängig voneinander verfahren, die Achsen werden miteinander interpoliert.

Istwert setzen G92

Löschen der G92–Verschiebung bei PowerOn:
24004: $MC_CHBFRAME_POWERON_MASK=1

G92 beleibt bei Reset (M30, Kanlreset) erhalten:
20110: $MC_RESET_MODE_MASK Bit0 und Bit14=1

Zurücksetzen des Werkzeugkoordinatensystems G92.1

Mit G92.1 X.. (G-Code-System A: G50.3 P0) kann ein verschobenes Koordinatensystem vor dem Verschieben zurückgesetzt werden. Damit wird das Werkstückkoordinatensystem auf das Koordinatensystem zurückgesetzt, welches durch die aktive einstellbaren Nullpunktverschiebungen (G54–G59) definiert ist. Ist keine einstellbare Nullpunktverschiebung aktiv, so wird das Werkstückkorrdinatensystem auf die Referenzposition gesetzt. G92.1 setzt Verschiebungen, die durch G92 oder G52 durchgeführt wurden, zurück. Zurückgesetzt werden aber nur die Achsen, die programmiert werden.

Beispiel 1:

```
N10 G0 X100 Y100      ;Anzeige: WCS: X100 Y100   MCS: X100 Y100
N20 G92 X10 Y10       ;Anzeige: WCS: X10 Y10     MCS: X100 Y100
N30 G0 X50 Y50        ;Anzeige: WCS: X50 Y50     MCS: X140 Y140
N40 G92.1 X0 Y0       ;Anzeige: WCS: X140 Y140   MCS: X140 Y140
```

Beispiel 2:

```
N10 G10 L2 P1 X10 Y10
N20 G0 X100 Y100      ;Anzeige: WCS: X100 Y100   MCS: X100 Y100
N30 G54 X100 Y100     ;Anzeige: WCS: X100 Y100   MCS: X110 Y110
N40 G92 X50 Y50       ;Anzeige: WCS: X50 Y50     MCS: X110 Y110
N50 G0 X100 Y100      ;Anzeige: WCS: X100 Y100   MCS: X160 Y160
N60 G92.1 X0 Y0       ;Anzeige: WCS: X150 Y150   MCS: X160 Y160
```

Restweg löschen

Bei ISO–Dialekt–Original wird mit G31 das Restweglöschen möglich gesetzt. Nur in diesem Satz wird der Restweg gelöscht, wenn das PLC Signal an den Kanal ansteht. Ohne G31 wird das Signal nicht ausgewertet.

Im ISO–Dialekt–Mode wird unabhängig von G31 das PLC Signale in jedem Satz ausgewertet. Mit G31 wird der Meßtaster1 aktiv gesetzt.

Bei ISO–Dialekt kann der gelöschte Restweg über PLC Varselektor errechnet werden.

Die Funktion G31 P1 (..P4) unterscheidet sich von G31 nur dadurch, daß mit P1 – P4 unterschiedliche Eingänge für das Meßsignal ausgewählt werden können. Dabei können auch mehrere Eingänge gleichzeitig auf eine steigende Flanke eines Meßsignals überwacht werden. Die Zuordnung der Eingänge zu den Adressen P1 –P4 wird über Maschinendaten festgelegt
10810: $MN_EXTERN_MEAS_G31_P_SIGNAL[0 .. 3].

Hinweis

Bei 840D stehen nur zwei Meßeingänge zur Verfügung.

Spindelposition

Die Spindelposition für M19 wird über das Settingdatum 43240 $SC_M19_SPOS gesetzt.

Schutzbereich

Werden die G–Befehle G22 und G23 benutzt, muß ein Schutzbereich vorhanden sein:

18190: $MM_NUM_PROTECT_AREA_NCK = 1
28210: $MC_NUM_PROTECT_AREA_ACTIVE = 1

Hilfsfunktionausgabe

Soll der Wert H als Integerwert an die PLC ausgegeben werden, ist
MD 22110: $MC_AUXFU_H_TYPE_INT auf 1 zu setzen.

Der Zeitpunkt der Hilfsfunktionsausgabe (M, S, T, H) an die PLC kann über MD eingestellt werden:

0 = Hilfsfunktionsausgabe vor der Bewegung

1 = Hilfsfunktionsausgabe während der Bewegung

2 = Hilfsfunktionsausgabe nach der Bewegung

3 = keine Ausgabe der Bewegung an die PLC

22200: $MC_AUXFU_M_SYNC_TYPE für M–Funktionen
22210: $MC_AUXFU_S_SYNC_TYPE für S–Funktionen
22220: $MC_AUXFU_T_SYNC_TYPE für T–Funktionen
22230: $MC_AUXFU_H_SYNC_TYPE für H–Funktionen

1. Referenzpunkt anfahren G28

Nachfolgende Maschinendaten müssen gesetzt sein:

20050: $MC_AXCONF_GEOAX_ASSIGN_TAB[0–2]
Achse 1 bis 3

20060 $MC_AXCONF_GEOAX_NAME_TAB[0–2]
Achsnamen für Fräsen: X, Y, Z
Achsnamen für Drehen: X, Z, Y

200070: $MC_AXCONF_MACHAX_USED[0–3]
Achse 1 bis 4

20080: $MC_AXCONF_CHANAX_NAME_TAB[0–3]
4. Achsname für Fräsen: X, Y, Z sind fest vorgegeben; zusätzlich kann A, B, C, U, V oder W ausgewählt werden.
4. Achsname für Drehen: X, Z, Y sind fest vorgegeben; zusätzlich kann C ausgewählt werden

20100: $MC_DIAMETER_AX_DEF
nur bei Drehen: X–Achse (in Vorbereitung)

20150: $MC_GCODE_RESET_VALUES[28]
Radius– oder Durchmesserprogrammierung
1 = DIAMOF (Radius bei G90/G91)
2 = DIAMON (Durchmesser bei G90/G91)
3 = DIAM90 (Durchmesser bei G90, Radius bei G91)
Hinweis: DIAM90 wird innerhalb des Zyklus wie DIAMON behandelt.

34100: $MA_REFP_SET_POS[0]
0 = 1. Referenzpunkt
Wert für jede Achse eingeben

35000: $MA_SPIND_ASSIGN_TO_MACHAX
0 = Achse ist keine Spindel
1 = Achse ist eine Spindel

2./3./4. Referenzpunkt anfahren G30

Nachfolgende Maschinendaten müssen gesetzt sein:

20050: $MC_AXCONF_GEOAX_ASSIGN_TAB[0–2]
Achse 1 bis 3

20060: $MC_AXCONF_GEOAX_NAME_TAB[0–2]
Achsnamen für Fräsen: X, Y, Z
Achsnamen für Drehen: X, Z, Y

20070: $MC_AXCONF_MACHAX_USED[0–3]
Achse 1 bis 4

20080: $MC_AXCONF_CHANAX_NAME_TAB[0–3]
4. Achsname für Fräsen: X, Y, Z sind fest vorgegeben; zusätzlich kann A, B, C, U, V oder W ausgewählt werden.
4. Achsname für Drehen: X, Z, Y sind fest vorgegeben; zusätzlich kann C ausgewählt werden

20100: $MC_DIAMETER_AX_DEF
nur bei Drehen: X–Achse (in Vorbereitung)

20150: $MC_GCODE_RESET_VALUES[28]
Radius– oder Durchmesserprogrammierung
1 = DIAMOF (Radius bei G90/G91)
2 = DIAMON (Durchmesser bei G90/G91)
3 = DIAM90 (Durchmesser bei G90, Radius bei G91)

Hinweis: DIAM90 wird innerhalb des Zyklus wie DIAMON behandelt.

34100: $MA_REFP_SET_POS[1,2,3]
1,2,3 = 2., 3., 4. Referenzpunkt
Wert für jede Achse eingeben

35000: $MA_SPIND_ASSIGN_TO_MACHAX
0 = Achse ist keine Spindel
1 = Achse ist eine Spindel

G30.1 floating reference Position

Referenzpunktverfahren wird im Zyklus CYCLE328 realisiert. Die Position des Referenzpunktes wird in das Settingdatum
43340: $SC_EXTERN_REF_POSITION_G30_1 geschrieben.

4.1.1 Aktiver G–Befehl an PLC

Über das MD 22512: $MC_EXTERN_GCODE_GROUPS_TO_PLC kann der Anwender die G–Gruppen einer externen Sprache auswählen, deren aktiver G–Befehl an die PLC gemeldet werden soll.

$MC_EXTERN_GCODE_GROUPS_TO_PLC[0..7]=0

4.1.2 Werkzeugwechsel, Werkzeugdaten

Bei Werkzeugwechsel wird keine Schneide angewählt.
20270: $MC_CUTTING_EDGE_DEFAULT = 0

Settingdatum: Bei Werkzeuganwahl wird mit G91 die Korrektur nicht herausgefahren
42442: $SC_TOOL_OFFSET_INCR_PROG = 0

Die Zuordnung der Werkzeuglängenkorrekturen zu den Geometrieachsen ist fest:
Länge 1: Z
Länge 2: Y
Länge 3: X
42940: $SC_TOOL_LENGTH_CONST = 17

Wirksamkeit der Werkzeuglängenkorrektur über Reset hinaus:
20110: $MC_RESET_MODE_MASK = ’B1000000’

Wirksamkeit der Werkzeugkorrektur mit Programmierung von T/H/D, nicht mit M6
22550: $MC_TOOL_CHANGE_MODE = 0

4.1.3 G00 immer mit Genauhalt

Bei hohen Geschwindigkeiten kann es bei aktivem Bahnsteuerbetrieb im G00 Mode durch das Überschleifen zu Kollisionen kommen. Über das Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK, Bit 4 wird das Genauhaltverhalten bei G00 bestimmt.

20734: $MC_EXTERN_FUNCTION_MASK, Bit 4=0, G00 wird mit der gerade aktiven Genauhaltfunktion verfahren. Ist G64 aktiv, werden die G00 Sätze auch mit G64 verfahren.

20734: $MC_EXTERN_FUNCTION_MASK, Bit 4=1, jeder G00 Satz mit Verfahrbewegung wird mit G09 (satzweiser Genauhalt) verfahren. Auch wenn G64 aktiv ist, wirkt der satzweise Genauhalt in jedem G00 Satz.

4.1.4 Verhalten bei syntaktischen Fehlern

Über das Maschinendatum 20734: $MC_EXTERN_FUNCTION_MASK, Bit 3 wird das Verhalten bei auftretenden Fehlern im ersten Teil des ISO Translators bestimmt. Dabei wird der komplette ASCII Satz untersucht.

Ist das Bit3==0 wird beim Finden von unbekannten Adressen ein NC–Alarm ausgegeben und die weitere Bearbeitung gestoppt.

Ist das Bit3==1 wird kein Alarm abgesetzt, der ASCII Satz wird an den Siemens Translator weitergegeben. Im Siemens Translator wird versucht, den Satz zu übersetzen, der danach folgende NC–Satz wird wieder zuerst an den ISO Translator gegeben.
Damit ist es möglich, eindeutige Siemenssätze zu programmieren ohne mit G290 in den Siemensmode zu wechseln, während der ISO–Mode aktiv ist.

Im GCode Fenster wird der aktuelle ISO GCode angezeigt, eine Umschaltung in den Siemensmode erfolgt nicht.
Wird in einem solchen Satz im Siemens Translator eine G–Funktion aktiviert, die direkt auf einen ISO GCode umgesetzt werden kann, erfolgt hier ein Update der GCodes.

Beispiel

20734: $MC_EXTERN_FUNCTION_MASK, Bit3==0

```
N5 G291          ;ISO Mode
N10 WAIT         ;Alarm 12080 “WAIT unbekannt”
N15 G91 G500     ;Alarm 12080 “G500 unbekannt”
```

20734: $MC_EXTERN_FUNCTION_MASK, Bit3==1

```
N5 G291          ;ISO Mode
N10 WAIT         ;Satz wird von Siemens Translator bearbeitet
N15 G91 G500     ;Satz wird von Siemens Translator bearbeitet
N20 X Y          ;Satz wird wegen G291 von ISO Translator bear–
                 ;beitet, G91 aus N15 ist aktiv
```

Hinweis

Durch Fehlprogrammierung im ISO Mode kann es zu ungewollten Reaktionen kommen.
Beispiel ISO M:

programmiert werden sollte	G90 G76	;modaler Zyklenaufruf
es wird aber	G90 G75	eingegeben

G75 gibt es im ISO–M–Mode nicht, der Satz geht an den Siemens Translator und führt dort ohne Abfrage oder Alarm G75 zu “Fahren auf Festpunkt”.

4.1.5 Anwahl Codesystem A, B, C (ISO–Dialekt–T)

Im ISO–Dialekt–T wird zwischen G Codesystem A, B, und C unterschieden. Defaultmäßig ist G Codesystem B aktiv. Die Umschaltung erfolgte bisher über einen Zyklus, der die Funktion “G Codes umbennen” über das Maschinendatum 10712: $MN_NC_USER_CODE_CONF_NAME_TAB nutzt.
Diese Methode hat den Nachteil, daß bei Erweiterungen des G Codes der Zyklus angepaßt werden muß, außerdem ist die Funktion “G Codes umbennen” für den Anwender nicht mehr nutzbar.

Softwarestand 6

Das Default G Codesystem ist weiterhin das G Codesystem B.

In das Maschinendatum 10881: $MN_MM_EXTERN_GCODE_SYSTEM wird Code System A, B oder C ausgewählt. Die Funktion “G Codes umbenennen” wird von dieser Funktion nicht genutzt, der Anwender kann diese Funktion also wieder uneingeschränkt einsetzen.

Auch das Umschalten mit einem Zyklus wie bisher ist weiterhin möglich.

$MN_MM_EXTERN_CNC_SYSTEM = 1: ISO–Dialekt–M
$MN_MM_EXTERN_CNC_SYSTEM = 2: ISO–Dialekt–T

$MN_MM_EXTERN_GCODE_SYSTEM = 0: G Codesystem B
$MN_MM_EXTERN_GCODE_SYSTEM = 1: G Codesystem A
$MN_MM_EXTERN_GCODE_SYSTEM = 2: G Codesystem C

Damit die Hüllzyklen im richtigen G-Codesystem arbeiten, muß das entsprechende System in die GUD Variable _ZSFI[39] eingetragen werden:

_ZSFI[39]: Settingdatum für G–Codesystem bei ISO-T
0 = G–Codesystem B
1 = G–Codesystem A
2 = G–Codesystem B
3 = G–Codesystem C

Inch/metrisch Umschaltung

Bei ISO–Dialekt–0riginal gibt es ein MD, mit dem bestimmt wird, wie die inch/metr. Umschaltung programmiert wird, entweder mit G20/21 oder G70/71. Dieses MD gibt es für den ISO–Dialekt–Mode nicht, die Auswahl ist auch nicht über $MN_MM_EXTERN_GCODE_SYSTEM möglich.

Defaultmäßig ist G20/G21 aktiv. Eine Umschaltung nach G70/71 wird mit dem 10712: MD $MN_USER_CODE_CONF_NAME_TAB realisiert werden.

$MN_USER_CODE_CONF_NAME_TAB[0]= G20
$MN_USER_CODE_CONF_NAME_TAB[1]= G70
$MN_USER_CODE_CONF_NAME_TAB[2]= G21
$MN_USER_CODE_CONF_NAME_TAB[3]= G71

Unabhängig von G20/21 G70/71 wird in der Systemvariable $P_GG[6] bei G20/G70 eine 1 und bei G21/G71 eine 2 gelesen.

4.1.6 Feste Vorschübe F0 – F9

Mit F0 – F9 können zehn verschiedene, über Settingdaten voreingestellte, Vorschubwerte aktiviert werden.
Um mit F0 Eilganggeschwindigkeit zu aktivieren, muß die entsprechende Geschwindigkeit in das Settingdatum
42160: $SC_EXTERN_FIXED_FEEDRATE_F1_9[0] eingetragen werden.

Die Vorschubwerte für F0 – F9 werden in den Settingdaten als Realwerte eingegeben. Eine Bewertung der Eingabewerte wird nicht vorgenommen.

Die Funktion wird über das Maschinendatum
22920: $MC_EXTERN_FIXED_FEEDRATE_F1_ON aktiviert. Ist das MD auf FALSE gesetzt, wird F1 – F9 als normale Vorschubprogrammierung interpretiert, z.B. F2=2 mm/min, F0=0 mm/min.
Ist das MD= TRUE, werden für F0 – F9 die Vorschubwerte aus den Settingdaten 42160: $SC_EXTERN_FIXED_FEEDRATE_F1_F9[ ] geholt. Steht in einem der Settingdaten der Wert 0, wird bei der Programmierung der korrespondierenden Adresserweiterung der Vorschub 0 aktiviert.

Beispiel

$SC_FIXED_FEEDRATE_F1_F9[0] = 5000
$SC_FIXED_FEEDRATE_F1_F9[1] = 1000
$SC_FIXED_FEEDRATE_F1_F9[2] = 500

```
N10 X10 Y10 Z10 F0 G94    ;Position mit 5000 mm/min anfahren
N20 G01 X150 Y30 F1       ;Vorschub 1000 mm/min aktiv
N30 Z0 F2                 ;Position wird mit 500 mm/min angefahren
N40 Z10 F0                ;Position mit 5000 mm/min anfahren
```

Ist die Funktion mit dem MD $MC_FIXED_FEEDRATE_F1_F9_ON aktiviert und soll mit F1 – F9 nicht der Vorschubwert aus dem Settingdatum aktiv werden, muß der Vorschubwert als Realwert programmiert werden. Soll ein Vorschubwert mit z.B. 1 mm/min programmiert werden, muß der Vorschub mit F1.0 statt F1 programmiert werden.

Hinweis

Bei Makroprogrammierung mit G65/66 wird für die Adresse F immer der programmierte Wert in die Zyklensystemvariable abgelegt. Bei F1 – F9 wird beispielsweise der Wert 1 – 9 in die Zyklensystemvariable $C_F eingetragen. Die Adresse hat hier die Bedeutung einer Übergabevariablen und keinen unmittelbaren Bezug zum Vorschub.

Das gleiche gilt für die Programmierung der Gewindesteigung bei G33 – G34 mit der Adresse F. Hier wird mit F kein Vorschub programmiert, sondern der Abstand zwischen 2 Gewindegängen bei einer Spindelumdrehung.

Bei der Zyklenprogrammierung (z.B. G81 X.. Y.. Z.. R.. P.. Q.. F..) wird unter der Adresse F immer der Vorschub programmiert. In einem Teileprogrammsatz mit einem Zyklenaufruf über eine G–Funktion (G81 – G87 usw.) wird deshalb bei der Programmierung von F1 – F9 der entsprechende Vorschubwert aus dem korrespondierenden Settingdatum in die Variable $C_F geschrieben.

Einschränkung

Bei ISO–Dialekt–Mode werden die Vorschubwerte in den Settingdaten mit einem Handrad verändert. Bei Siemensmode können die Vorschübe nur wie ein direkter programmierter Vorschub, z.B. über den Overrideschalter beeinflußt werden.

4.1.7 Parallele Achsen G17<Achsname>.. (G18 / G19)

Mit der Funktion G17 (G18, G19)<Achsname>.. kann eine zur Basisachse im Koordinatensystem parallele Achse aktiviert werden.

Die Basisachsen sind z.B.X, Y und Z. Mit der Programmierung

```
G17 U0 Y0     ;Aktivierung der parallelen Achse U
```

wird statt der Basisachse X, die Achse U in der G17 Ebene aktiv.

Diese Funktion kann mit der 840D Funktion GEOAX (..,..) nachgebildet werden. Mit GEOAX() kann eine Geometrieachse mit einer beliebigen Kanalachse getauscht werden. Dabei werden aber alle Frames (außer Handrad– und externe Verschiebung), die Arbeitsfeldbegrenzung und die Schutzbereiche gelöscht. Das Löschen der Frames kann mit den MD 10602: $MN_FRAME_GEOAX_CHANGE_MODE, das Löschen der Schutzbereiche mit dem MD 10618: $MC_PROTAREA_GEOAX_CHANGE_MODE und das Deaktivieren der Arbeitsfeldbegrenzung mit dem neuen MD 10604: $MN_WALIM_GEOAX_CHANGE_MODE verhindert werden.

Für jede der 3 Geometrieachsen kann je eine parallele Achse über das MD 22930: $MC_EXTERN_PARALLEL_GEOAX[ ] definiert werden. Mit der Programmierung der G–Funktion zur Ebenenanwahl (G17 – G19) und dem Achsnamen der parallelen Achse wird dann ein Geoachstausch, analog zu der Funktion GEOAX(), durchgeführt.

Die Achsen werden bei der Ebenenanwahl (Geoachstausch mit paralleler Achse) auf ihre programmierte Position gefahren.

Ist in einem Ebenenanwahlsatz eine Basisachse des Koordinatensystems zusammen mit ihrer parallelen Achse programmiert, wir der Alarm 12726 “Unzulässige Ebenenanwahl mit parallelen Achsen” ausgegeben.

4.1.8 Einfügen von Fasen und Radien

Das Einfügen von Fasen und Radien wird auf der entsprechenden Siemensfunktionalität abgebildet. Es müssen immer die beide Sätze programmiert werden, zwischen denen ein Radius oder eine Fase eingefügt werden soll. Werden mehrere Adressen in einem Satz programmiert, wirkt immer der zuletzt programmierte Radius.

Bei ISO–Dialekt–M Mode ist der Name für den Radius immer “R” und für die Fase immer “C”. Sobald ein Komma im Satz programmiert ist, werden die rechts nach dem Komma stehenden Adressen R und C als Radius und Fase interpretiert. Soll ein Kreis mit den Radius R oder die Achse C programmiert werden, müssen die beiden Adressen links vor dem Komma stehen. Es kann zwischen Linearsätzen, Kreissätzen oder einer Mischung aus beiden ein Radius oder eine Fase eingefügt werden.

Im ISO–Dialekt–T Mode ist der Name für den Radius immer “R”, für die Fase können die Adressen “C”, “I” und “K” verwendet werden. Die Adresse C darf nur verwendet werden, wenn sie nicht als Achsname definiert ist. Radien und Fasen können nur zwischen Linearsätzen eingefügt werden. Die Linearsätze müssen nicht senkrecht aufeinander stehen. Die Programmierung eines Vorzeichens vor der Fase bzw. Radius hat hier keine Bedeutung, da mit dem 2. Geradensatz die Richtung der Fase bzw. des Radius festgelegt ist.

Programmierung ISO–Dialekt–M

Fasen und Radien werden im Satz immer mit einem Komma markiert. Die Adresse für eine Fase ist “C”, für einen Radius “R”. Fasen und Radien können zwischen Linearsätzen und Kreissätzen eingefügt werden.

```
N10 X100. ,R10                  ;Radius von 10 mm einfügen
N20 Y30. ,C5                    ;Fase von 5 mm einfügen
N30 X150. Y40.
N40 G03 X180. Y65. R30 ,R8
N50 G01 X150. ,R8
```

Programmierung ISO–Dialekt–T

Fasen und Radien werden im Satz nicht mit einem Komma markiert. Die Adresse für einen Radius ist immer “R”, für eine Fase kann die Adresse “I”, “K” oder “C” sein. C darf nur verwendet werden, wenn die Adresse nicht als Achsname definiert ist.
Im ISO–Dialekt–T Mode können Fasen und Radien nur zwischen 2 Linearsätzen eingefügt werden.

```
N10 X100. R10                ;Radius von 10 mm einfügen
N20 Z30. C5                  ;Fase von 5 mm einfügen
N30 X150. Z40.
N40 X180. Z65. I8
N50 G01 X150. K8
```

4.1.9 Rundachsfunktion

Ist eine Achse als Rundachse definiert, kann diese Achse folgendermaßen verfahren werden:

- Achse wird verfahren wie eine Linearachse
  es ist keine Modulowandlung aktiv
  es können positive und negative Werte programmiert werden
  30300: $MA_IS_ROT_AX = TRUE
  30310: $MA_ROT_IS_MODULO = FALSE

- Beispiel: N5 G90 B0
  N10 B370 ; Achse fährt auf 370 Grad
  N15 B–10 ; Achse fährt auf –10 Grad
  N20 G91 B–20 ; Achse fährt auf –20 Grad

- Achse verfährt auf kürzestem Weg
  Modulowandlung ist aktiv
  es können Werte <0 und > Modulowert programmiert werden
  30300: $MA_IS_ROT_AX = TRUE
  30310: $MA_ROT_IS_MODULO = TRUE
  20754: $MC_EXTERN_FUNCTION_MASK, Bit5 = 0

30455: $MA_MISC_FUNCTION_MASK, Bit0 = 1

```
Beispiel:   N5 B0          ; Achse fährt auf 0 Grad
            N10 B10        ; Achse fährt auf 10 Grad, positive
                           ; Drehrichtung
            N15 B350       ; Achse fährt auf 350 Grad, negative
                           ; Drehrichtung
            N20 B–5        ; Achse fährt auf 355 Grad, positive
                           ; Drehrichtung
            N25 G91 B–10   ; Achse fährt auf 345 Grad, negative
                           ; Drehrichtung
```

- Achse verfährt abhängig vom programmierten Vorzeichen in positiver oder negativer Verfahrrichtung
  Modulowandlung ist aktiv
  es können Werte <0 und > Modulowert programmiert werden
  30300: $MA_IS_ROT_AX = TRUE
  30310: $MA_ROT_IS_MODULO = TRUE
  20754: $MC_EXTERN_FUNCTION_MASK, Bit5 = 1
  30455: $MA_MISC_FUNCTION_MASK, Bit0 = 1

```
Beispiel:   N5 B0          ; Achse fährt auf 0 Grad
            N10 B10        ; Achse fährt auf 10 Grad, positive
                           ; Drehrichtung
            N15 B350       ; Achse fährt auf 350 Grad, positive
                           ; Drehrichtung
            N20 B–5        ;Achse fährt auf 355 Grad, negative
                           ; Drehrichtung
            N25 G91 B–10   ; Achse fährt auf 345 Grad, negative
                           ; Drehrichtung
```

  In diesem Fall hat das Vorzeichen zwei Bedeutungen, es wird in der Modulowandlung berücksichtigt und bestimmt die Drehrichtung.
  Das Verhalten ist bei allen Interpolationsarten gleich.
  Inkrementelle Bewegungen werden immer abhängig vom Vorzeichen ausgeführt.

4.1.10 Programmkoordinierung zwischen 2 Kanälen und M-Befehlen

Um den Programmablauf zwischen zwei Kanälen zu synchronisieren, können M-Funktionen programmiert werden, die als WAIT-Marken dienen. Wird im Teileprogramm des einen Kanals eine solche M-Funktion erreicht, wird der Programmablauf solange gestoppt, bis auch der andere Kanal die selbe M–Funktion erreicht hat. Anschließend werden die Teileprogramme in beiden Kanälen weiter bearbeitet.

Die M-Funktionsnummern für die Wait-Marken werden mit zwei Maschinendaten definiert. Es wird dabei ein Bereich von M-Nummern festgelegt, der für diese Funktion reserviert ist.

Das Maschinendatum 10800: $MN_EXTERN_CHAN_SYNC_M_NO_MIN markiert die kleinste M-Nummer und
10802: $MN_EXTERN_CHAN_SYNC_M_NO_MAX die grösste M-Nummer des Bereichs, der für die Programmkoordinierung reserviert ist. Das Maschinendatum $MN_EXTERN_CHAN_SYNC_M_NO_MAX darf nicht größer als ($MN_EXTERN_CHAN_SYNC_M_NO_MIN + 10 * Kanalanzahl) sein.

Um Konflikte mit Standard-M-Funktionen zu vermeiden, darf die kleinste M-Nummer (MD $MN_EXTERN_CHAN_SYNC_M_NO_MIN) nicht kleiner als 100 sein. Die Vorbesetzung der Maschinendaten mit –1 bedeutet, daß keine Programmkoordinierung im ISO-Dialekt-T/M-Mode möglich ist.

Wird in $MN_EXTERN_CHAN_SYNC_M_NO_MIN ein Wert <100 oder in $MN_EXTERN_CHAN_SYNC_M_NO_MAX ein Wert <$MN_ETERN_CHAN_SYNC_M_NO_MIN eingetragen, wird der Alarm ”unzulässige-Nummer für Kanalsynchronisation” ausgegeben.

Die Funktion wird auf den WAITM-Befehl der Siemenssprache abgebildet (WAITM(<Marke>, <Kanalnummer>, <Kanalnummer>)). Dabei wird die Kanalsynchronisation im ISO-Dialekt-T/M–Mode immer für die Kanäle 1 und 2 durchgeführt. Alle anderen Kanäle können nur im Siemens–Mode synchronisiert werden.

Die M-Nummern werden nicht an die PLC ausgegeben.

Die M-Funktionen für die Kanalsynchronisation müssen alleine im Satz stehen. Werden weitere Adressen außer ”M” im Satz programmiert, wird der Alarm 12080 (Syntaxfehler) ausgegeben.

Nähere Beschreibung siehe

/PGA/ SINUMERIK 840D/810D/FM–NC Programmieranleitung Arbeitsvorbereitung, Kapitel: Programmkoordinierung.

Einschränkung

Es können nur 10 M-Funktionen (WAIT-Marken) je Kanal gesetzt werden. Die Differenz $MN_EXTERN_CHAN_SYNC_M_NO_MAX zu $MN_EXTERN_CHAN_SYNC_M_NO_MIN darf deshalb in einem 2–Kanalsystem nicht größer als 20 sein. Bei ISO-Dialekt-Original können 99999899 MNummern als Waitmarken definiert werden.

4.2 Standardbelegung der Maschinendaten für ISO–Dialekt

ISO–Dialekt–M

Standardbelegung des MD 20154: $MC_EXTERN_GCODE_RESET_VALUES[ ]:

$MC_EXTERN_GCODE_RESET_VALUES[0]=1	G00
$MC_EXTERN_GCODE_RESET_VALUES[1]=1	G17
$MC_EXTERN_GCODE_RESET_VALUES[2]=1	G90
$MC_EXTERN_GCODE_RESET_VALUES[3]=2	G23
$MC_EXTERN_GCODE_RESET_VALUES[4]=1	G94
$MC_EXTERN_GCODE_RESET_VALUES[5]=1	G20
$MC_EXTERN_GCODE_RESET_VALUES[6]=1	G40
$MC_EXTERN_GCODE_RESET_VALUES[7]=3	G49
$MC_EXTERN_GCODE_RESET_VALUES[8]=4	G80
$MC_EXTERN_GCODE_RESET_VALUES[9]=1	G98
$MC_EXTERN_GCODE_RESET_VALUES[10]=1	G50
$MC_EXTERN_GCODE_RESET_VALUES[11]=2	G67
$MC_EXTERN_GCODE_RESET_VALUES[12]=2	G97
$MC_EXTERN_GCODE_RESET_VALUES[13]=1	G54
$MC_EXTERN_GCODE_RESET_VALUES[14]=3	G64
$MC_EXTERN_GCODE_RESET_VALUES[15]=2	G69
$MC_EXTERN_GCODE_RESET_VALUES[16]=1	G15
$MC_EXTERN_GCODE_RESET_VALUES[17]=0	satzweise
$MC_EXTERN_GCODE_RESET_VALUES[21]=1	G50.1
$MC_EXTERN_GCODE_RESET_VALUES[24]=1	G12.1
$MC_EXTERN_GCODE_RESET_VALUES[30]=1	G290

ISO–Dialekt–T

Standardbelegung des MD 20154: $MC_EXTERN_GCODE_RESET_VALUES[ ]:

Für den ISO–Dialekt–T werden optional mehrer G–Code–Systeme angeboten. Dabei wird die gleiche Funktion über unterschiedliche G–Befehle aufgerufen. Standardmäßig wird das G–Code–System B realisiert. Zur Nutzung eines anderen G–Code–Systems wird das Maschinendatum
10882: $MN_NC_USER_EXTERN_GCODES_TAB eingeführt (siehe 4.2.1)

$MC_EXTERN_GCODE_RESET_VALUES[0]=1	G00
$MC_EXTERN_GCODE_RESET_VALUES[1]=2	G97
$MC_EXTERN_GCODE_RESET_VALUES[2]=1	G90
$MC_EXTERN_GCODE_RESET_VALUES[3]=2	G69
$MC_EXTERN_GCODE_RESET_VALUES[4]=2	G95
$MC_EXTERN_GCODE_RESET_VALUES[5]=1	G21
$MC_EXTERN_GCODE_RESET_VALUES[6]=1	G40
$MC_EXTERN_GCODE_RESET_VALUES[8]=2	G23
$MC_EXTERN_GCODE_RESET_VALUES[9]=1	G80
$MC_EXTERN_GCODE_RESET_VALUES[10]=1	G98
$MC_EXTERN_GCODE_RESET_VALUES[11]=2	G67
$MC_EXTERN_GCODE_RESET_VALUES[15]=2	G18
$MC_EXTERN_GCODE_RESET_VALUES[17]=0	satzweise
$MC_EXTERN_GCODE_RESET_VALUES[19]=1	G50.2
$MC_EXTERN_GCODE_RESET_VALUES[20]=1	G12.1
$MC_EXTERN_GCODE_RESET_VALUES[31]=1	G290

■

Randbedingungen

5

Verfügbarkeit der Funktion ”ISO–Dialekt”

Die Funktion ist eine Option und verfügbar bei

- SINUMERIK 810D mit CCU1 und CCU2
- SINUMERIK 840D mit NCU 572.2 und NCU 573.2

5.1 Einschränkungen

Im folgenden sind Funktionen aufgeführt bei denen im ISO–Dialekt–Mode der SINUMERIK 840D ein **nicht kompatibles** Verhalten gegenüber dem ISO–Dialekt–Original auftritt.

Mode–Umschaltung

Der Standard–Maschinendatensatz realisiert nur den “Siemens”–Mode. Es wird keine externe NC–Sprache als 2. G–Code–Tabelle generiert.

Das Maschinendatum 10712: $MN_NC_USER_CODE_CONF_NAME_TAB gilt nur für NC–Sprachbefehle im Siemens–Mode.

Aus Kompatibilitätsgründen zum Siemens–Mode (betrifft Maschinendateneingabe, BTSS–Schnittstelle: “Datenfeld”[0] = 1. G–Gruppe) werden im ISO–Dialekt–Mode die im System ISO–Dialekt–Original definierten G–Gruppen 0 umgesetzt:

ISO–Dialekt–M: G–Gruppe 0 → G–Gruppe 18
ISO–Dialekt–T: G–Gruppe 0 → G–Gruppe 17

Implizierter Modewechsel

Asups, INI–Files und Makro/GUD–Definitionsfiles werden immer im Siemensmode bearbeitet. Gegebenenfalls wird hierzu ein impliziter Modewechsel in den Siemens–Mode durchgeführt. Mit dem Ende der Bearbeitung wird wieder in den ursprünglichen Mode des externen CNC–Systems zurückgeschaltet.

5.1.1 Programmbefehle

F–Wert

ISO–Dialekt–M hat feste F–Werte, welche mit F1 bis F9 angewählt werden. Diese festen Werte gibt es im ISO–Dialekt–Mode nicht. F1 bis F9 wird als Wert F1 bis F9 interpretiert.

G02/G03

Die Programmierung G02/G03 ohne Radiusangabe führt bei ISO–Dialekt–T–Original zu G01, bei ISO–Dialekt–M–Original zu Vollkreis mit undefiniertem Radius. Im ISO–Dialekt–Mode wird in beiden Fällen ein NC Alarm abgesetzt.

G04 X..

Verweilzeit, bei ISO–Dialekt–Original wird die Verweilzeit in der X–Achse als Restweg angezeigt. Diese Verweilzeit wird im ISO–Dialekt–Mode nicht angezeigt, es erscheint die Meldung “Verweilzeit läuft”.

G16

Polarkoordinatenprogrammierung wird mit G15 beendet.
Bei ISO–Dialekt–Original bleibt der Polradius und Polwinkel erhalten. Bei der nächsten Programmierung von G16 kann inkrementell auf den Winkel aufgesetzt werden. Dabei wird eine nicht nachvollziehbare Verfahrbewegung ausgeführt. Der Winkel und der Radius wird mit Reset oder M30 gelöscht.

Im ISO–Dialekt–Mode wird mit G15 der Polradius und der Polwinkel gelöscht. Mit G16 wird immer mit einem Winkel und Radius von 0 gefahren.

Beispiel: Achse U liegt parallel zu Achse X
G17 U Y ; Ebene U Y ist angewählt, statt X Y Ebene.

Im ISO–Dialekt–Mode kann bei G17/G18/G19 **keine** parallele Achse programmiert werden.

G20/G21

Im ISO–Dialekt–Original–Mode werden bei der Umschaltung die Nullpunktverschiebungen nicht umgerechnet. Es wird lediglich das Komma um eine Zehnerstelle verschoben. Im ISO–Dialekt–Mode erfolgt eine vollständige Umrechnung der Nullpunktverschiebungswerte.

G22

Im ISO–Dialekt–M–Original wird mit G22 ein Schutzbereich 4 aktiviert. Diesen gibt es nicht in Schritt 1. Es werden die Schutzbereiche 1/2 und 3 realisiert. Schutzbereich 4 wird bei ISO–Dialekt über Maschinendaten fest eingegeben. Das ist bei 840D nicht möglich. Im ISO–Dialekt–Mode gibt es nur einen Schutzbereich.

G40

Im Mode ISO–Dialekt–T kann im Linearsatz ein Vektor mit I, J, K programmiert werden, der das Verhalten am Satzende beeinflusst. Diese Funktion ist nicht im ISO–Dialekt–Original–Mode möglich. Wird bei G40 I, J und K programmiert, kommt es zu einem NC–Alarm.

G41/G42

Die Funktionen Fräserradiuskorrektur sind nicht kompatibel zwischen ISO–Dialekt–Original und 840D.

G53

Wird G53 (Position im Maschinenkoordinatensystem anfahren) während G41/G42 aktiv ist aufgerufen, werden die Achsen nicht getrennt, sondern interpolierend verfahren.

G63

Bei ISO–Dialekt kann G63 in jedem Satz geschrieben werden. In diesem Satz ist der Override gesperrt. Override 0 führt auch am Anfang eines G53–Satzes zu Halt. Sinnvoll ist diese Funktion in Verbindung mit Gewindebohren ohne Ausgleichsfutter in einer G01 Interpolation. Im ISO–Dialekt–Mode wirkt G63 nur in einem G01 Satz. Wenn G63 in einem G00 Satz angewählt wird hat das auf diesen Satz keine Auswirkung.

G94/G95

Wird von Umdrehungsvorschub (G95) nach Linearvorschub (G94) oder umgekehrt gewechselt, so ,muß jedesmal der Vorschub F neu programmiert werden. Fehlt der Vorschub, führt dies zum Alarm 10860 ”Kein Vorschub programmiert”. ISO–Dialekt–Original–Mode interpretiert den einmal programmierten Vorschub bei einem Wechsel der Vorschubart entweder als Umdrehungs– oder Linearvorschub.

G96

Bei ISO–Dialekt–M kann die Achse, auf die sich G96 bezieht, mit P programmiert werden. Dies führt im ISO–Mode zu einem Alarm. Möglich ist nur P0 Planachse aus Maschinendaten.

M06

Werkzeugwechselzyklus: Wird über MD 10715: $MN_M_NO_FCT_CYCLE und 10716: $MN_M_NO_FCT_CYCLE_NAME einer M–Funktion ein Zyklus zugeordnet, kann im Zyklus nicht auf alle im Satz programmierten Parameter zugegriffen werden. Es kann nur die programmierte T–Nummer mit $C_T/$C_T_PROG gelesen werden. Verfahrbewegungen im M–Satz werden vor dem Aufruf herausgefahren.

Syntaxbestimmende G–Funktionen

Bei ISO–Dialekt–Original können mehrere syntaxbestimmende G–Befehle in einem Satz programmiert werden. Es wird unterschieden nach:

syntaxbestimmend → satzweise

syntaxbestimmend → modal

nichtsyntax → modal

nichtsyntax → satzweise

Im Siemens–Mode führt mehr als eine syntaxbestimmende G–Funktion im Satz zu einem Alarm.

5.1.2 Werkzeugverwaltung

Werkzeugverwaltung, Standzeit– und Stückzahlüberwachung können mit der Siemens–Werkzeugverwaltung nachgebildet werden.

Werkzeugdaten

Fräsen: Es wird nur Tool Compensation Memory C unterstützt, d.h. mehrspaltiger Aufbau des Werkzeugkorrkturspeichers (es gilt D == H).
Die Variante, daß Werkzeug– und Korrekturnummer beim Fräsen aus demselben Wert gebildet werden, wird nicht unterstützt.

Das Auslesen/Archivieren der aktuellen Werkzeugdaten im ISO–Dialekt–Mode (G10) ist nicht möglich. Verändern der Werkzeugdaten über G10 ist erst möglich, wenn die Werkzeugkorrekturen per Bedienung eingerichtet wurden.

Werkzeuglängenkorrektur

Ist die Werkzeuglängenkorrektureingabe im Durchmesser aktiv, so kann bei ISO–Dialekt–Original die Eingabe für Geometrie und Verschleiß im Durchmesser parametriert werden. Im ISO–Dialekt–Mode wird nur der Verschleiß im Durchmesser angegeben. Die Geometrie wird immer im Radius angegeben.

Bei ISO–Dialekt–Original wird abhängig von einem Maschinenparameter die Werkzeuglänge als Durchmesser oder Radius in der Planachse verrechnet.
Im Siemens–Mode wird die Längenkorrektur immer als Durchmesser verrechnet; eine Umschaltung ist nicht möglich.

Das Kompensieren der Werkzeuglängenkorrektur durch Verschieben des Koordinatensystems ist nicht möglich.

Bei der Technologie Drehen können Geometrie und Verschleiß nicht aus verschiedenen Korrekturspeichern genommen werden.

Geänderte Werkzeugkorrekturen werden mit der Programmierung des nächsten T–, H– oder D–Wertes gültig.

5.1.3 Steuerungsverhalten bei Power On, Reset und Satzsuchlauf

Power On

Bei Power On wird für alle NC–Kanäle die Siemens–G–Code–Liste mit den möglichen Umcodierungen aus dem MD 10712:
$MN_NC_USER_CODE_CONF_NAME_TAB angelegt.

Hochlauf, Reset

Das Hochlauf– und Resetverhalten wird über MD 20150:
$MC_GCODE_RESET_VALUES[46] und in Abhängigkeit des MD 10880:
$MN_MM_EXTERN_CNC_SYSTEM festgelegt.

Dieses entscheidet zwischen den G–Codes von ISO–Dialekt–M und ISO–Dialekt–T.

Der Wechsel zwischen Siemens– und ISO–Dialekt–Mode im Programmablauf hat keinen Einfluß auf die modalen G–Funktionen.

Satzsuchlauf

Der im Siemens–Mode vorhandene Satzsuchlauftyp “Suchlauf auf Satzendpunkt” entspricht dem Suchlaufverhalten bei ISO–Dialekt.

In der Betriebsart “Satzsuchlauf ohne Berechnungen” muß der Anwender, besonders bei NC–Programmen mit Mode–Umschaltung, für ein sinnvolles Suchziel sorgen (z.B. NC–Satz mit Befehl aus G–Gruppe 47).

■

Platz für Notizen

Datenbeschreibungen (MD, SD) 6

6.1 Allgemeine Maschinendaten

10604 SD–Nummer	**WALIM_GEOAX_CHANGE_MODE** Arbeitsfeldbegrenzung beim Umschalten von Geoachsen	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 1
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 6.2	
Bedeutung:	Mit diesem Maschinendatum wird festgelegt, ob beim Geoachstausch eine eventuell aktive Arbeitsfeldbegrenzung erhalten bleibt oder deaktiviert wird. Das MD ist bitcodiert mit folgenden Bedeutungen: Bit 0= =0: Arbeitsfeldbegrenzung wird bei Geoachstausch deaktiviert =1: Aktive Arbeitsfeldbegrenzung bleibt bei Geoachstausch aktiviert	

10615 MD–Nummer	**NCBFRAME_POWERON_MASK** Globale Basisframes bei Power On löschen	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 0
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 5.2	
Bedeutung:	Mit diesem Maschinendatum wird festgelegt, ob globale Basisframes bei Power On Reset gelöscht werden. Die Anwahl kann für die einzelnen Basisframes getrennt erfolgen. Bit 0 entspricht Basisframe 0, Bit 1 Basisframe 1 usw. 0: Basisframe bleibt bei Power On erhalten 1: Basisframe wird bei Power On gelöscht.	

10652	**CONTOUR_DEF_ANGLE_NAME**			
MD–Nummer	Einstellbarer Name für Winkel in der Kontur–Kurzbeschreibung			
Standardvorbesetzung: ”ANG”		min. Eingabegrenze: –		max. Eingabegrenze: –
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: STRING			gültig ab SW–Stand: 5	
Bedeutung:	Die Einstellung wirkt nur für Siemens–G–Code–Programmierung, d.h. G290. Der Name, unter dem der Winkel in der Kontur–Kurzbeschreibung programmiert wird, ist einstellbar. Damit kann z.B. identische Programmierung in verschiedenen Sprachmodi realisiert werden: Wird als Name “A” angegeben, so wird der Winkel in der Siemens–Programmierung wie bei ISO–Dialekt angegeben. Der Bezeichner muß eindeutig sein, d.h. es duerfen keine gleichnamigen Achsen, Variable, Makros usw. existieren.			

10654	**RADIUS_NAME**			
MD–Nummer	Einstellbarer Name für Radius satzweise in der Kontur–Kurzbeschreibung			
Standardvorbesetzung: ”RND”		min. Eingabegrenze: –		max. Eingabegrenze: –
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: STRING			gültig ab SW–Stand: 5	
Bedeutung:	Der Name, unter dem der Radius in der Kontur–Kurzbeschreibung programmiert wird, ist einstellbar. Damit kann z.B. identische Programmierung in verschiedenen Sprachmodi realisiiert werden: Wird als Name “R” angegeben, so wird der Radius in der Siemens–Programmierung wie bei ISO–Dialekt angegeben. Der Bezeichner muß eindeutig sein, d.h. es dürfen keine gleichnamigen Achsen, Variable, Makros usw. existieren. Die Einstellung wirkt für Siemens–G–Code–Programmierung, d.h. G290.			

10656	**CHAMFER_NAME**			
MD–Nummer	Einstellbarer Name für Fase in der Kontur–Kurzbeschreibung			
Standardvorbesetzung: ”CHR”		min. Eingabegrenze: –		max. Eingabegrenze: –
Änderung gültig nachPOWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: STRING			gültig ab SW–Stand: 5	
Bedeutung:	Der Name, unter dem die Fase in der Kontur–Kurzbeschreibung programmiert wird, ist einstellbar. Damit kann z.B. identische Programmierung in verschiedenen Sprachmodi realisiiert werden: Wird als Name “C” angegeben, so wird der Radius in der Siemens–Programmierung wie bei ISO–Dialekt angegeben. Der Bezeichner muß eindeutig sein, d.h. es dürfen keine gleichnamigen Achsen, Variable, Makros usw. existieren. Die Einstellung wirkt für Siemens–G–Code–Programmierung, d.h. G290. Die Fase wirkt in ursprünglicher Bewegungsrichtung. Alternativ kann die Fasenlänge unter dem Bezeichner CHF programmiert werden.			

10704	**DRYRUN_MASK**	
MD–Nummer	Aktivierung des Probelaufvorschubs	
Standardvorbesetzung:	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nach	Schutzstufe:	Einheit: –
Datentype: BYTE	gültig ab SW–Stand:	
Bedeutung:	DRYRUN_MASK == 0 Dryrun darf nur am Satzende ein- und ausgeschaltet werden. DRYRUN_MASK == 1 wird die De-/Aktivierung des Probelaufvorschubs auch während einer Programmbearbeitung möglich sein Achtung: Nach der Aktivierung des Probelaufvorschubs wird für die Dauer des Reorganisierungsvorgangs die Achsen gestoppt. DRYRUN_MASK == 2 Dryrun ist in jeder Phase ein- und ausschaltbar und die Achsen werden nicht gestoppt. **Achtung:** Allerdings wird die Funktion erst mit einem im Programmablauf ”späteren” Satz wirksam! Mit dem nächsten (impliziten) StopRe-Satz wird die Funktion wirksam.	

10706	**SLASH_MASK**	
MD–Nummer	Aktivierung der Satzausblendung	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 2
Änderung gültig nach	Schutzstufe:	Einheit: –
Datentype: BYTE	gültig ab SW–Stand:	
Bedeutung:	SLASH_MASK == 0 Satzausblendung umschalten ist nur am Satzende gestoppt möglich. SLASH_MASK == 1 Bei SLASH_MASK == 1 wird die Aktivierung der Satzausblendung auch während einer Programmbearbeitung möglich sein. Achtung: Nach der Aktivierung der Satzausblendung wird für die Dauer des Reorganisierungsvorgangs die Achsen gestoppt. SLASH_MASK == 2 Satzumschaltung ist in jeder Phase möglich. **Achtung:** Allerdings wird die Funktion erst mit einem im Programmablauf ”späteren” Satz wirksam! Mit dem nächsten (impliziten) StopRe-Satz wird die Funktion wirksam.	

<table>
<tr><td>10715</td><td colspan="3">M_NO_FCT_CYCLE[0]</td></tr>
<tr><td>MD–Nummer</td><td colspan="3">M–Funktionsnummer für Zyklenaufruf</td></tr>
<tr><td>Standardvorbesetzung: –1</td><td>min. Eingabegrenze: –1</td><td colspan="2">max. Eingabegrenze: –</td></tr>
<tr><td>Änderung gültig nach POWER ON</td><td>Schutzstufe: 2/7</td><td colspan="2">Einheit: –</td></tr>
<tr><td colspan="2">Datentype: DWORD</td><td colspan="2">gültig ab SW–Stand: 5.2</td></tr>
<tr><td>Bedeutung:</td><td colspan="3">M–Nummer mit der ein Unterprogramm aufgerufen wird.

Der Name des Unterprogramms steht in $MN_M_NO_FCT_CYCLE_NAME. Wird in einem Teileprogramm die mit $MN_M_NO_FCT_CYCLE festgelegte M–Funktion programmiert, wird am Satzende das in M_NO_FCT_CYCLE_NAME definierte Unterprogramm gestartet.

Wird die M–Funktion im Unterprogramm nochmals programmiert, findet die Ersetzung durch einen Unterprogrammaufruf nicht mehr statt.

$MN_M_NO_FCT_CYCLE wirkt sowohl im Siemens–Mode G290, als auch im externen Sprach–Mode G291.

M–Funktionen mit fester Bedeutung dürfen nicht mit einem Unterprogrammaufruf überlagert werden.
Im Konfliktfall wird dies mit Alarm 4150 gemeldet:
– M0 bis M5,
– M17, M30,
– M40 bis M45,
– M–Funktion zur Umschaltung Spindelbetrieb/Achsbetrieb laut $MC_SPIND_RIGID_TAPPING_M_NR (Vorbelegung M70)
– M–Funktionen für Nibbeln/Stanzen laut Projektierung über $MC_NIBBLE_PUNCH_CODE sofern sie über $MC_PUNCHNIB_ACTIVATION aktiviert wurden.
– bei applizierter externer Sprache ($MN_MM_EXTERN_LANGUAGE) M19, M96–M99.

Ausnahme: die mit $MC_TOOL_CHANGE_M_CODE festgelegten M–Funktionen für den Werkzeugwechsel.

Die mit $MN_M_NO_FCT_CYCLE_NAME und $MN_T_NO_FCT_CYCLE_NAME projektierten Unterprogramme dürfen nicht gleichzeitig in einem Satz (Teileprogrammzeile) wirksam werden, d.h. pro Satz kann maximal eine M/T–Funktionsersetzung wirksam werden. In dem Satz mit der M–Funktionsersetzung darf weder ein M98– noch ein modaler Unterprogramm–Aufruf programmiert sein. Auch Unterprogrammrücksprung und Teileprogrammende sind nicht erlaubt.
Im Konfliktfall wird Alarm 14016 abgesetzt.</td></tr>
</table>

10716	**M_NO_FCT_CYCLE_NAME[0]**	
MD–Nummer	Name für Werkzeugwechselzyklus bei M–Funktionen aus MD $MN_MFCT_CYCLE	
Standardvorbesetzung: –	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nachPOWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: STRING	gültig ab SW–Stand: 5.2	

Bedeutung:

Im Maschinendatum steht der Name des Zyklus. Dieser Zyklus wird aufgerufen, wenn die M–Funktion aus dem Maschinendatum $MN_M_NO_FCT_CYCLE programmiert wurde.
Ist die M–Funktion in einem Bewegungssatz programmiert, so wird der Zyklus nach der Bewegung ausgeführt.

$MN_M_NO_FCT_CYCLE wirkt sowohl im Siemens–Mode G290, als auch im externe Sprach–Mode G291.
Ist im Aufrufsatz eine T–Nummer programmiert, so kann die programmierte T–Nummer im Zyklus unter der Variablen $P_TOOL abgefragt werden.

$MN_M_NO_FCT_CYCLE_NAME und $MN_T_NO_FCT_CYCLE_NAME dürfen nicht gleichzeitig in einem Satz wirksam werden, d.h. pro Satz kann maximal eine M–/T–Funktionsersetzung wirksam werden. In dem Satz mit der M–Funktionsersetzung darf weder ein M98– noch ein modaler Unterprogramm–Aufruf programmiert sein. Auch Unterprogrammrücksprung und Teileprogrammende sind nicht erlaubt.
Im Konfliktfall wird Alarm 14016 abgesetzt.
Die Adreßerweiterung der M–Funktion kann im Zyklus über die Systemvariable $C_ME abgefragt werden.
Beispiel:

```
$MN_M_NO_FCT_CYCLE = 6
$MN_M_NO_FCT_CYCLE_NAME = ”MSUB”

PROC MAIN
...
N100 M[2]=6                    ;Adreßerweiterung 2
...
M30

PROC MSUB
...
N200 M[$C_ME]=$C_M
...
```

Die Adresserweiterung zur programmierten T–Nummer kann im Zyklus über die Systemvariable $C_TE abgefragt werden.
Beispiel:

```
$MN_T_NO_FCT_CYCLE_NAME = ”TSUB”

PROC MAIN
...
N100 T[1]=6                    ;Adreßerweiterung 1
...
M30

PROC TSUB
...
N200 T[$C_ME]=$C_M
...
```

Die programmierte T–Nummer kann im Zyklus über die Systemvariablen
$C_T bzw. $C_T_PROG als Dezimalwert und über
$C_TS bzw. $C_TS_PROG als String (nur mit Werkzeugverwaltung)
abgefragt werden.

Wird mit der T–Nummer auch eine D–Nummer programmiert, kann diese auch im Zyklus über die Systemvariablen $C_D bzw. $C_D_PROG abgefragt werden.

10716	**M_NO_FCT_CYCLE_NAME[0]**
MD–Nummer	Name für Werkzeugwechselzyklus bei M–Funktionen aus MD $MN_MFCT_CYCLE
Bedeutung:	Mit der Systemvariable $C_DL_PROG kann im Zyklus abgefragt werden, ob mit der T–Funktion auch die Adresse DL (Summenkorrektur) programmiert wurde. Der programmierte Wert kann dann über die Systemvariable $C_DL gelesen werden. Beispiel: $MN_T_NO_FCT_CYCLE_NAME = ”TSUB” PROC MAIN ... ;Werkzeugwechsel mit N100 T5 D1 DL=2 ;Schneidenanwahl und Summenkorrektur ... M30 PROC TSUB ... N190 IF $SC_DL_PROG == 1 N200 DL=$SC_DL N210 ENDIF

10717	**T_NO_FCT_CYCLE_NAME**			
MD–Nummer	Name für Werkzeugwechselzyklus bei T–Nummer			
Standardvorbesetzung: –		min. Eingabegrenze: –		max. Eingabegrenze: –
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: STRING			gültig ab SW–Stand: 5.2	
Bedeutung:	Zyklenname für Werkzeugwechselroutine bei Aufruf über T–Funktion. Wird in einem Teileprogrammsatz eine T–Funktion programmiert, so wird am Satzende, das in T_NO_FCT_CYCLE_NAME definierte Unterprogramm aufgerufen. Die programmierte T–Nummer kann im Zyklus über die Systemvariablen $C_T / $C_T_PROG als Dezimalwert und über $C_TS / $C_TS_PROG als String (nur mit Werkzeugverwaltung) abgefragt werden. Wird mit der T–Nummer auch eine D–Nummer programmiert, kann diese im Zyklus über die Systemvariablen $C_D/$C_D_PROG abgefragt werden. Im Unterprogramm kann über die Systemvariable $C_T_PROG bzw. $C_D_PROG abgefragt werden, ob T bzw. D programmiert wurde. Mit den Systemvariablen $C_T bzw. $C_D können die Werte ausgelesen werden. Wird im Unterprogramm erneut ein T–Befehl programmiert, wird keine Ersetzung ausgeführt, sondern das T–Wort an die PLC ausgegeben. $MN_T_NO_FCT_CYCLE_NAME und die Systemvariablen $C_T / $C_TS_PROG wirkt sowohl im Siemens–Mode G290, als auch im externen Sprach–Mode G291. $MN_M_NO_FCT_CYCLE_NAME und $MN_T_NO_FCT_CYCLE_NAME dürfen nicht gleichzeitig in einem Satz wirksam werden, d.h. pro Satz kann maximal eine M/T–Funktionsersetzung wirksam werden. In dem Satz mit der T–Funktionsersetzung darf weder ein M98– noch ein modaler Unterprogramm–Aufruf programmiert sein. Auch Unterprogrammrücksprung und Teileprogrammende sind nicht erlaubt. Im Konfliktfall wird Alarm 14016 abgesetzt. Die Adreßerweiterung zur programmierten T–Nummer kann im Zyklus über die Systemvariable $C_TE abgefragt werden. Beispiel: $MN_T_NO_FCT_CYCLE_NAME = ”TSUB” PROC MAIN ... N100 T[1]=5 ;Adreßerweiterung 2 ... M30 PROC TSUB ... N200 M[$C_TE]=$C_T ...			

10718	M_NO_FCT_CYCLE_PAR			
MD–Nummer	M–Funktionsersetzung mit Parametern			
Standardvorbesetzung: –1		min. Eingabegrenze: –	max. Eingabegrenze: –	
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: DWORD			gültig ab SW–Stand: 6.3	
Bedeutung:	Wurde mit MD 10715: M_NO_FCT_CYCLE[n] / MD 10716: M_NO_FCT_CYCLE_NAME[n] eine M–Funktionsersetzung projektiert, so kann mit MD 10718: M_NO_FCT_CYCLE_PAR für eine dieser M–Funktionen eine Parameterübergabe per Systemvariable wie bei der T–Funktionsersetzung spezifiziert werden. Die in den Systemvariablen abgelegten Parameter beziehen sich immer auf die Teileprogrammzeile in der die zu ersetzenden M–Funktion programmiert wurde. Folgende Systemvariable stehen zur Verügung: $C_ME : Adresserweiterung der substituierten M–Funktion $C_T_PROG : TRUE wenn Adresse T programmiert wurde $C_T : Wert der Adresse T (Integer) $C_TE : Adresserweiterung der Adresse T $C_TS_PROG : TRUE wenn Adresse TS programmiert wurde $C_TS : Wert der Adresse TS (String, nur mit Werkzeugverwaltung) $C_D_PROG : TRUE wenn Adresse D programmiert wurde $C_D : Wert der Adresse D $C_DL_PROG : TRUE wenn Adresse DL programmiert wurde $C_DL : Wert der Adresse DL			

10719	T_NO_FCT_CYCLE_MODE			
MD–Nummer	Parametrierung der T–Funktions–Ersetzung			
Standardvorbesetzung: 0		min. Eingabegrenze: –	max. Eingabegrenze: –	
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: DWORD			gültig ab SW–Stand: 6.4	
Bedeutung:	Mit diesem Maschinendatum wird eingestellt, ob bei gleichzeitiger Programmierung von D bzw. DL und T in einem Satz D bzw. DL als Parameter an den T–Ersetzungszyklus überge- ben wird (Voreinstellung) oder vor dem Aufruf des T–Ersetzungszyklusses ausgeführt wer- den soll. Wert 0: Wie bisher, die D– bzw. DL–Nummer wird an den Zyklus übergeben (Default–Wert) Wert 1: die D– bzw. DL–Nummer wird direkt im Satz verrechnet Diese Funktion ist nur aktiv, wenn der Werkzeugwechsel mit M–Funktion projektiert wurde (MD 22550: TOOL_CHANGE_MODE = 1), anderweitig werden die D– bzw. DL–Werte immer übergeben.			

10760	**G53_TOOLCORR**	
MD–Nummer	Wirkungsweise bei G53, G153 und SUPA	
Standardvorbesetzung: 2	min. Eingabegrenze: 2	max. Eingabegrenze: 4
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses MD wirkt im Siemens–Mode und im externen Sprachmode. Mit diesem Maschinendatum wird festgelegt, ob bei den Sprachbefehlen G53, G153 und SUPA Werkzeuglängen– und Werkzeugradiuskorrektur unterdrückt werden soll. 0 = G53/G153/SUPA ist ein satzweises Unterdrücken von Nullpunktverschiebungen, aktive Werkzeuglängen– und Werkzeugradiuskorrektur bleibt erhalten. 1= G53/G153/SUPA ist ein satzweises Unterdrücken von Nullpunktverschiebungen und aktive Werkzeuglängen– und Werkzeugradiuskorrektur.	

10800	**EXTERN_CHAN_SYNC_M_NO_MIN**	
MD–Nummer	Erste M–Nummer für Kanalsynchronisation	
Standardvorbesetzung: –1	min. Eingabegrenze: 100	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 6.2	
Bedeutung:	Kleinste M–Nummer des M–Nummernbereiches der für die Kanalsynchronisation reserviert ist.	

10802	**EXTERN_CHAN_SYNC_M_NO_MAX**	
SD–Nummer	Letzte M–Nummer für Kanalsynchronisation	
Standardvorbesetzung: –1	min. Eingabegrenze: 100	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 6.2	
Bedeutung:	Größte M–Nummer des M–Nummernbereichs, der für die Kanalsynchronisation reserviert ist. Der M–Nummernbereich darf maximal 10*Kanalanzahl groß sein (bei 2 Kanälen = 20 M–Nummern). Wird ein größerer Bereich festgelegt, wird der Alarm 4170 ausgegeben.	

10804	**EXTERN_M_NO_SET_INT**	
MD–Nummer	M–Funktion für ASUP Aktivierung	
Standardvorbesetzung: 96	min. Eingabegrenze: 0	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 6.2	
Bedeutung:	M–Funktionsnummer, mit der im ISO_T/M–Mode ein Interruptprogramm aktiviert wird (ASUP).	

10806	EXTERN_M_NO_DISABLE_INT			
MD–Nummer	M–Funktion für ASUP Deaktivierung			
Standardvorbesetzung: 97		min. Eingabegrenze: 0		max. Eingabegrenze:
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: DWORD			gültig ab SW–Stand: 6.2	
Bedeutung:	M–Funktionsnummer, mit der im ISO–T/M–Mode ein Interruptprogramm deaktiviert wird (ASUP).			

10808	EXTERN_INTERRUPT_BITS_M96			
MD–Nummer	Interruptprogramm–Bearbeitung (M96)			
Standardvorbesetzung: 0		min. Eingabegrenze: 0		max. Eingabegrenze: 8
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: WORD			gültig ab SW–Stand: 6.2	
Bedeutung:	Mit dem Setzen der verschiedenen Bits kann der Ablauf der mit M96 P.. aktivierten Interruptroutine beeinflußt werden. Bit 0: =0, Kein Interruptprogramm möglich, M96/M97 sind normale M–Funktionen =1, Aktivierung eines Interruptprogramms mit M96/M97 erlaubt Bit 1: =0, Teileprogramm mit der Endposition des nächsten Satzes nach dem Unterbrechungssatz weiterbearbeiten =1, Teileprogramm ab der Unterbrechungposition weiterbearbeiten Bit 2: =0, das Interruptsignal unterbricht den aktuellen Satz sofort und startet die Interruptroutine =1, die Interruptroutine wird erst am Ende des Satzes gestartet Bit 3: =0, Bearbeitungszyklus bei einem Interruptsignal unterbrechen =1, Interruptprogramm erst am Ende des Bearbeitungszyklus starten			

10810	EXTERN_MEAS_G31_P_SIGNAL			
MD–Nummer	Zuordnung der Meßeingänge für G31 P..			
Standardvorbesetzung: 1		min. Eingabegrenze: 0		max. Eingabegrenze: 3
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: BYTE			gültig ab SW–Stand: 6.2	
Bedeutung:	Mit dem Maschinendatum wird eine Zuordnung der Meßeingänge 1 und 2 zu den mit G31 P1 (–P4) programmierten P–Nummern festgelegt. Das MD ist Bit-codiert. Es werden nur Bit 0 und Bit 1 ausgewertet. Ist z.B. in $MN_EXTERN_MEAS_G31_P_SIGNAL[1] das Bit 0=1, wird mit G31 P2 der 1. Meßeingang aktiviert. Mit $MN_EXTERN_MEAS_G31_P_SIGNAL[3] = 2 wird mit G31 P4 der 2. Meßeingang aktiviert. Bit 0: =0: Meßeingang 1 bei G31 P1 (–P4) nicht auswerten =1 Meßeingang 1 bei G31 P1 (–P4) aktivieren Bit 1: =0 Meßeingang 2 bei G31 P1 (–P4) nicht auswerten =1 Meßeingang 2 bei G31 P1 (–P4) aktivieren			

10812	**EXTERN_DOUBLE_TURRET_ON**		
MD–Nummer	Doppelrevolverkopf mit G68		
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:	
Änderung gültig nach	Schutzstufe:	Einheit: –	
Datentype: BOOLEAN	gültig ab SW–Stand: 6.2		
Bedeutung:	Das Maschinendatum ist nur wirksam bei $MN_EXTER_CNC_SYSTEM = 2. Turning Mit diesem MD wird festgelegt, ob mit G68 eine Doppelschlittenbearbeitung (Kanalsynchronisation für 1. und 2. Kanal) gestartet werden soll oder das zweite Werkzeug eines Doppelrevolvers (= 2, mit dem im Settingdatum $SC_EXTERN_DOUBLE_TURRET_DIST definierten Abstand, fest miteinander verbundene Werkzeug) aktiviert werden soll. FALSE: Kanalsynchronisation für Doppelschlittenbearbeitung TRUE: 2. Werkzeug eines Doppelrevolvers einwechseln (= $SC_EXTERN_DOUBLE_TURRET_DISTANCE als additive Nullpunktverschiebung und Spiegeln um Z-Achse aktivieren		

10814	**EXTERN_M_NO_MAC_CYCLE**	
MD–Nummer	Makroaufruf über M–Funktion	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand:	
Bedeutung:	M–Nummer, mit der ein Makro aufgerufen wird. Der Name des Unterprogramms steht in $MN_EXTERN_M_NO_MAC_CYCLE_NAME[n]. Wird in einem Teileprogrammsatz die mit $MN_EXTERN_M_NO_MAC_CYCLE[n] festgelegte M–Funktion programmiert, wird das im EXTERN_M_NO_MAC_CYCLE_NAME[n] definierte Unterprogramm gestartet, alle im Satz programmierten Adressen werden in die dazugehörige Variablen geschrieben. Wird die M–Funktion im Unterprogramm nochmals programmiert, findet die Ersetzung durch einen Unterprogrammaufruf nicht mehr statt. $MN_EXTERN_M_NO_MAC_CYCLE_NAME[n] wirkt nur im externen Sprach–Mode G291. M–Funktionen mit fester Bedeutung dürfen nicht mit einem Unterprogrammaufruf überlagert werden. Im Konfliktfall wird dies mit Alarm 4150 gemeldet: – M0 bis M5, – M17, M30, – M19, – M40 bis M45, – M–Funktion zur Umschaltung Spindelbetrieb/Achsbetrieb laut $MC_SPIND_RIGID_TAPPING_M_NR (vorbelegung: M70), – M–Funktionen für Nibbeln/Stanzen laut Projektierung über $MC_NIBBLE_PUNCH_CODE sofern sie über $MC_PUNCHNIB_ACTIVATION aktiviert wurden. – bei applizierter externer Sprache ($MN_MM_EXTERN_LANGUAGE) zusätzlich M96 bis M99 – M–Funktionen, die durch $MN_M_NO_FCT_CYCLE defiiert sind. Ausnahme: Die mit $MC_TOOL_CHANGE_M_CODE festgelegte M–Funktion für den Werkzeugwechsel. Die mit $MN_EXTERN_M_NO_MAC_CYCLE_NAME[n] projektierten Unterprogramme dürfen nicht gleichzeitig in einem Satz (Teileprogrammzeile) wirksam werden, d.h. pro Satz kann maximal eine M–Funktionsersetzung wirksam werden. In dem Satz mit der M–Funktionsersetzung darf weder ein M98– noch ein modaler Unterprogramm–Aufruf programmiert sein. Auch Unterprogrammrücksprung und Teileprogrammende sind nicht erlaubt. Im Konfliktfall wird der Alarm 14016 abgesetzt.	

10815	**EXTERN_M_NO_MAC_CYCLE_NAME**	
MD–Nummer	UP–Name für M–Funktion Makroaufruf	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe:	Einheit: –
Datentype: STRING	gültig ab SW–Stand:	
Bedeutung:	Zyklenname bei Aufruf über die mit $MN_EXTERN_M_NO_MAC_CYCLE[n] definierte M–Funktion.	

10816	**EXTERN_G_NO_MAC_CYCLE**	
MD–Nummer	Makroaufruf über G–Funktion	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe:	Einheit: –
Datentype: DOUBLE	gültig ab SW–Stand: 6.3	
Bedeutung:	G–Nummer mit der ein Makro aufgerufen wird. Der Name des Unterprogramms steht in $MN_EXTERN_G_NO_MAC_CYCLE_NAME[n]. Wird in einem Teileprogrammsatz die mit $MN_EXTERN_G_NO_MAC_CYCLE[n] festgelegte G–Funktion programmiert, wird das in EXTERN_M_NO_MAC_CYCLE_NAME[n] definierte Unterprogramm gestartet, alle im Satz programmierten Adressen werden in die dazugehörigen $C_xx Variablen geschrieben. Ist bereits ein Unterprogrammaufruf über ein M/G–Makro oder eine M–Substitution aktiv, wird kein Unterprogrammaufruf ausgeführt. Ist in diesem Fall eine Standard–G–Funktion programmiert, wird diese ausgeführt, anderenfalls wird der Alarm 12470 ausgegeben. $MN_EXTERN_G_NO_MAC_CYCLE[n] wirkt nur im externen Sprach–Mode G291. In einem Satz darf nur ein Unterprogrammaufruf stehen. D.h. in einem Satz darf immer nur eine M/G–Funktionsersetzung programmiert werden und es darf kein zusätzlicher Unterprogramm(M98)– oder Zyklenaufruf im Satz stehen. Auch Unterprogrammrücksprung und Teileprogrammende im selben Satz sind nicht erlaubt. Im Konfliktfall wird Alarm 14016 abgesetzt.	

10817	**EXTERN_G_NO_MAC_CYCLE_NAME**	
MD–Nummer	UP–Name für G–Funktion Makroaufruf	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe:	Einheit: –
Datentype: STRING	gültig ab SW–Stand: 6.3	
Bedeutung:	Zyklenname bei Aufruf über die mit $MN_EXTERN_G_NO_MAC_CYCLE[n] definierte G–Funktion.	

10818	**EXTERN_INTERRUPT_NUM_ASUP**	
MD–Nummer	Interruptnummer für ASUP–Start (M96)	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe:	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 6	
Bedeutung:	Nummer des Interrupteingangs, mit dem ein im ISO–Mode aktiviertes asynchrones Unterprügramm gestartet wird. (M96< Programmnummer>)	

10820	**EXTERN_INTERRUPT_NUM_RETRAC**	
MD–Nummer	Interruptnummer für Schnellrückzug (G10.6)	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig nach POWER ON	Schutzstufe:	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 6	
Bedeutung:	Nummer des Interrupteingangs, mit dem im ISO–Mode ein Schnellrückzug auf die mit G10.6 programmierte Position ausgelöst wird.	

10880	**MM_EXTERN_CNC_SYSTEM**	
MD–Nummer	Externes Steuerungssystem, dessen Programme abgearbeitet werden	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 2
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: WORD	gültig ab SW–Stand: 5	
Bedeutung:	Auswahl der externen Sprache 1 = ISO–2: System Fanuc0 Milling (ab 5.1) 2 = ISO–3: System Fanuc0 Turning (ab 5.2) Dabei gilt der in den aktuellen Siemens–Dokumentationen definierte Funktionsumfang. Dieses Datum wird nur bei gesetztem Maschinendatum $MN_MM_EXTERN_LANGUAGE ausgewertet.	

10881	**MM_EXTERN_GCODE_SYSTEM**	
SD–Nummer	ISO Mode T: GCode System	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 2
Änderung gültig nach POWER ON	Schutzstufe:2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 6.2	
Bedeutung:	Festlegung des GCodeSystems, das im ISO–Dialekt–T Mode aktiv abgearbeitet werden soll: Wert = 0: ISO_T: Code System B Wert = 1: ISO_T: Code System A Wert = 2: ISO_T: Code System C Damit die Hüllzyklen im richtigen G–Codesystem arbeiten, muß das entsprechende System in die GUD Variable _ZSFI[39] eingetragen werden:	

10882	**NC_USER_EXTERN_GCODES_TAB [n]:0...59**	
MD–Nummer	Liste anwenderspezifischer G–Befehle einer externen NC–Sprache	
Standardvorbesetzung: –	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nach POWER ON	Schutzstufe: 2/2	Einheit: –
Datentype: STRING	gültig ab SW–Stand: 5	
Bedeutung:	Standardmäßig ist für externe Programmiersprache ISO–Dialekt–T Code B realisiert. Code A und C unterscheiden sich durch die Namen der G–Funktionen. Über $MN_NC_USER_EXTERN_GCODES_TAB kann die Umbennenung der G–Funktionen erfolgen. G–Befehle für externe NC–Sprachen können umcodiert werden. Die G–Gruppe und die Stellung innerhalb der G–Gruppe bleiben erhalten. Es dürfen nur G–Befehle umcodiert werden. Maximal sind 30 Umcodierungen möglich. Beispiel: $MN_NC_USER_EXTERN_GCODES_TAB[0]=”G20” $MN_NC_USER_EXTERN_GCODES_TAB[1]=”G70” —> G20 wird in G70 umcodiert; Ist G70 bereits vorhanden, erscheint bei NCK–Reset eine Fehlermeldung.	

10884	EXTERN_FLOATINGPOINT_PROG				
MD–Nummer	Bewertung programmierter Werte ohne Dezimalpunkt				
Standardvorbesetzung: 1		min. Eingabegrenze: 0		max. Eingabegrenze: 1	
Änderung gültig nach POWER ON			Schutzstufe: 2/7		Einheit: –
Datentype: BOOLEAN				gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses Maschinendatum wirkt für externe Programmiersprachen, d.h. wenn MD 18800: MM_EXTERN_LANGUAGE = 1. Mit diesem Maschinendatum wird festgelegt wie programmierte Werte ohne Dezimalpunkt bewertet werden. 0: Standard Notation: Werte ohne Dezimalpunkt werden in interne Einheiten IS–B, IS–C interpretiert (siehe MD EXTERN_INCREMENT_SYSTEM). Werte ohne Dezimalpunkt werden in interne Einheiten interpretiert z. B. X1000 = 1 mm (bei 0.001 mm Eingabefeinheit) X1000.0 = 1000 mm 1: PocketCalculator Notation: Werte ohne Dezimalpunkt werden als mm, inch oder Grad interpretiert. Werte ohne Dezimalpunkt werden als mm, inch oder Grad interpretiert z.B. X1000 = 1000 mm X1000.0 = 1000 mm				

10886	EXTERN_INCREMENT_SYSTEM				
MD–Nummer	Inkrementsystem				
Standardvorbesetzung: 0		min. Eingabegrenze: 0		max. Eingabegrenze: 1	
Änderung gültig nach POWER ON			Schutzstufe: 2/7		Einheit: –
Datentype: BOOLEAN				gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses Maschinendatum wirkt für externe Programmiersprachen, d.h. wenn MD 18800: MM_EXTERN_LANGUAGE = 1. Mit diesem Maschinendatum wird festgelegt welches Inkrement–System aktiv ist 0: Inkrementsystem IS–B = 0.001 mm/Grad = 0.0001 inch 1: Inkrementsystem IS–C = 0.0001 mm/Grad = 0.00001 inch				

10888	EXTERN_DIGITS_TOOL_NO				
MD–Nummer	Stellenanzahl für T–Nummer in externem Sprachemode				
Standardvorbesetzung: 2		min. Eingabegrenze: 2		max. Eingabegrenze: 4	
Änderung gültig nach POWER ON			Schutzstufe: 2/7		Einheit: –
Datentype: BYTE				gültig ab SW–Stand: 5.2	
Bedeutung:	Das Maschinendatum ist nur wirksam bei $MN_EXTERN_CNC_SYSTEM = 2. Stellenanzahl Werkzeugnummer im programmierten T–Wert. Aus dem programmierten T–Wert werden die über $MN_EXTERN_DIGITS_TOOL_NO angegebene Anzahl führender Stellen als Werkzeugnummer interpretiert. Die folgenden Stellen adressieren den Korrekturspeicher.				

10890	**EXTERN_TOOLPROG_MODE**	
MD–Nummer	Werkzeugwechselprogrammierung bei externer Programmiersprache	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 1
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 5.2	
Bedeutung:	Konfiguration der Programmierung des Werkzeugwechsels bei externer Programmiersprache: Bit0 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Im T–Wert werden Werkzeugnummer und Korrekturnummer programmiert. $MN_DIGITS_TOOLNO bstimmt die Anzahl der führenden Stellen, die die Werkzeugnummer bilden. Beispiel: $MN_DIGITS_TOOL_NO = 2 T=1234 ; Werkzeugnummer 12, ; Korrekturnummer 34 Bit0 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Im T–Wert wird nur die Werkzeugnummer programmiert. Korrekturnummer = Werkzeugnummer. $MN_DIGITS_TOOL_NO ist irrelevant. Beispiel: T=12 ; Werkzeugnummer 12 ; Korrekturnummer 12 Bit1 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Ist die Anzahl der im T–Wert programmierten Stellen gleich der in $MN_EXTERN_DIGITS_TOOL_NO bestimmten Anzahl, so werden führende 0 ergänzt Bit1 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Ist die Anzahl der im T–Wert programmierten Stellen gleich der in $MN_EXTERN_DIGITS_TOOL_NO angegebenen Stellenanzahl, so gilt die programmierte Nummer als Korrektur– und Werkzeugnummer Bit2 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: ISO T Korrekturanwahl nur mit D (Siemens Schneidennummer) Bit2 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: ISO T Korrekturanwahl nur mit H ($TC_DPH[t,d]) Bit3 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Es ist pro TOA jede H–Nummer nur einmal erlaubt, außer H=0. Wird das Bit3 1 –> 0 gesetzt, darf keine H–Nummer in einer TO–Einheit mehrfach vorkommen. Es kommt sonst beim nächsten Warmstart zu einem Alarm. Bit3 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 2: Es ist pro TOA jede H–Nummer mehrmals erlaubt. Bit6 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 1: Anwahl Werkzeuglänge unter der Adresse H nicht möglich Bit6 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 1: Anwahl Werkzeuglänge unter der Adresse H Bit7 = 0: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 1: Anwahl Werkzeuglänge unter der Adresse D nicht möglich Bit7 = 1: Wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 1: Anwahl Werkzeuglänge unter der Adresse D Sind Bit6 und Bit7 gesetzt, so ist die Anwahl unter Adresse D oder H möglich.	

18800	MM_EXTERN_LANGUAGE		
MD–Nummer	Externe Sprache in der Steuerung aktiv		
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 1	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –	
Datentype: DWORD	gültig ab SW–Stand: 5		
Bedeutung:	Zur Abarbeitung von Teileprogrammen anderer Steuerungshesteller muß die entsprechende NC–Sprache aktiviert werden. Es ist nur eine externe Sprache auswählbar. Der jeweils bereit gestellte Befehlsumfang ist den aktuellen Dokumentatinen zu etnehmen. Bit 0 (LSB): Abarbeitung von Teileprogrammen ISO_2 oder ISO_3. Codierung siehe $MN_MM_EXTERN_CNC_SYSTEM (10880)		

6.2 Kanalspezifische Maschinendaten

20094	SPIND_RIGID_TAPPING_M_NR		
MD–Nummer	M–Nummer für die Umschaltung in den gesteuerten Spindelbetrieb (Siemens–Mode)		
Standardvorbesetzung: 70	min. Eingabegrenze: 0	max. Eingabegrenze: 0xFF	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –	
Datentype: BYTE	gültig ab SW–Stand: 5.2		
Bedeutung:	Das Maschinendatum wirkt im Siemens–Mode sowie im externen Sprachmode. Mit diesem Maschinendatum wird festgelegt, mit welcher M–Funktionsnummer die Spindel in den gesteuerten Spindelbetrieb (Achsbetrieb) umgeschaltet wird. Mit dieser Nummer wird im Siemensmode M70 und im externen Sprachmode M29 ersetzt. Es sind nur M–Nummern erlaubt, die nicht schon als Standard festgelegt sind. Nicht erlaubt sind z. B. die M–Nummern M1, M2, M3, M4, M5 M30 usw.		

20095	EXTERN_RIGID_TAPPING_M_NR		
MD–Nummer	M–Nummer für die Umschaltung in den gesteuerten Spindelbetrieb (externer Sprachmode)		
Standardvorbesetzung: 29	min. Eingabegrenze: 6	max. Eingabegrenze: 0xFF	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –	
Datentype: BYTE	gültig ab SW–Stand:		
Bedeutung:	Mit diesem Maschinendatum wird im externen Sprachmodus festgelegt, mit welcher M–Funktionsnummer die Spindel in den gesteuerten Spindelbetrieb (Achsbetrieb) umgeschaltet wird. Mit dieser Nummer kann im externen Sprachmode M29 durch eine andere M–Funktion ersetzt werden. Es sind nur M–Nummern erlaubt, die nicht schon als Standard festgelegt sind. Nicht erlaubt sind z.B. die M–Nummern M0, M1, M3, M4, M5, M30, M99 usw.		

20152	**GCODE_RESET_MODE**	
MD–Nummer	Resetverhalten der G-Gruppen	
Standardvorbesetzung:	min. Eingabegrenze: 0	max. Eingabegrenze: 1
Änderung gültig nach RESET	Schutzstufe: 2/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand:	
Bedeutung:	Dieses Maschinendatum wird nur bei gesetztem Bit 0 in $MC_RESET_MODE_MASK ausgewertet. Mit desem MD wird für jeden Eintrag im MD $MN_GCODE_RESET_VALUES (also für jede G-Gruppe) festgelegt, ob bei einem Reset/Teileprogrammende wieder die Einstellung entsprechend $MC_GCODE_RESET_VALUES genommen wird (MD = 0) oder die momentan aktuelle Einstellung erhalten bleibt (MD = 1). Beispiel: Hier wird bei jedem Reset/Teileprogrammende die Grundstellung für die 6. G-Gruppe (aktuelle Ebene) aus dem MD $MC_GCODE_RESET_VALUES gelesen: $MC_GCODE_RESET_VALUE(5)=1; Resetvalue der 6. G-Gruppe ist M17 $MC_GCODE_RESET_MODE(5)=0; Grundstellung für 6. G-Gruppe ist nach Reset/Teileprogrammende entsp. $MC_GCODE_RESET_VALUES(5) Soll die aktuelle Einstellung für die 6. G-Gruppe (altuelle Ebene) jedoch über Reset/Teil programmende hinaus erhalten bleiben, so ergibt sich folgende Einstellung: $MC_GCODE_RESET_VALUE(5)=1; Resetvalue der 6. G-Gruppe ist M17 $MC_GCODE_RESET_MODE(5)=1; aktuelle Einstellung für 6. G-Gruppe bleibt auch nach Reset/Teileprogrammende erhalten	

20154	**EXTERN_GCODE_RESET_VALUES[n]: 0, ..., 30**	
MD–Nummer	Festlegung der G–Codes, die im Hochlauf wirksam werden, wenn der NC–Kanal nicht im Siemens–Mode läuft.	
Standardvorbesetzung: –	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nach POWER ON	Schutzstufe: 2/2	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 5	
Bedeutung:	Folgende externe Programmiersprachen sind möglich: – ISO–Dialekt–Milling – ISO–Dialekt–Turning Die zu verwendende G–Gruppen–Einteilung ergibt sich aus den aktuellen SINUMERIK–Dokumentationen. Folgende Gruppen innerhalb des MD EXTERN_GCODE_RESET_VALUES sind schreibbar: **ISO–Dialekt–M:** G–Gruppe 2: G17/G18/G19 G–Gruppe 3: G90/G91 G–Gruppe 5: G94/G95 G–Gruppe 6: G20/G21 G–Gruppe 13: G96/G97 G–Gruppe 14: G54–G59 **ISO–Dialekt–T:** G–Gruppe 2: G96/G97 G–Gruppe 3: G90/G91 G–Gruppe 5: G94/G95 G–Gruppe 6: G20/G21 G–Gruppe 16: G17/G18/G19	

20380	**TOOL_CORR_MODE_G43/G44**		
MD–Nummer	Behandlung der Werkzeuglängenkorrektur G43/G44		
Standardvorbesetzung: 0	min. Eingabegrenze: 1		max. Eingabegrenze: 2
Änderung gültig nach RESET	Schutzstufe: 2/7		Einheit: –
Datentype: BYTE		gültig ab SW–Stand: 5.2	
Bedeutung:	Das Maschinendatum wirkt nur bei $MN_MM_EXTERN_CNC_LANGUAGE = 1; Es bestimmt bei aktivem G43/G44 die Art wie mit H programmierte Längenkorrekturen verarbeitet werden. 0: Modus A Die Werkzeuglänge H wirkt immer auf die Z–Achse, unabhängig von der aktuellen Ebene 1: Modus B Die Werkzeuglänge H wirkt abhängig von der aktiven Ebene auf eine der drei Geometrieachsen und zwar bei G17 auf die 3. Geometrieachse (in der Regel Z) G18 auf die 2. Geometrieachse (in der Regel Y) G19 auf die 1. Geometrieachse (in der Regel X) In diesem Modus können durch mehrfache Programmierung Korrekturen in allen drei Geometrieachsen aufgebaut werden, d.h. durch die Aktivierung einer Komponente wird die in einer anderen Achse eventuell bereits wirksame Längenkorrektur nicht gelöscht. 2: Modus C Die Werkzeuglänge wirkt unabhängig von der aktiven Ebene in der Achse, die gleichzeitig mit H programmiert wurde. Im übrigen ist das Verhalten wie bei der Variante B		

20382	**TOOL_CORR_MOVE_MODE**		
MD–Nummer	Herausfahren der Werkzeuglängenkorrektur		
Standardvorbesetzung: FALSE	min. Eingabegrenze: –		max. Eingabegrenze: –
Änderung gültig nach RESET	Schutzstufe: 2/7		Einheit: –
Datentype: BOOLEAN		gültig ab SW–Stand: 5.2	
Bedeutung:	Das Maschinendatum bestimmt, wie die Werkzeuglängenkorrekturen herausgefahren werden. FALSE: Eine Werkzeuglängenkomponente wird nur herausgefahren, wenn die zugehörige Achse programmiert wurde (Verhalten wie in bisherigen Softwareständen) TRUE: Werkzeuglängen werden immer sofort herausgefahren, unabhängig davon, ob die zugehörigen Achsen programmiert sind oder nicht.		

20732	**EXTERN_G0_LINEAR_MODE**		
MD–Nummer	Interpolationsverhalten bei G00		
Standardvorbesetzung: 1	min. Eingabegrenze: 0		max. Eingabegrenze: 1
Änderung gültig nach POWER ON	Schutzstufe: 2/4		Einheit: –
Datentype: BOOLEAN		gültig ab SW–Stand:	
Bedeutung:	Mit diesem MD wird das Interpolationsverhalten bei G00 festgelegt. 0: Achsen werden als Positionierachsen verfahren 1: Achsen interpolieren miteinander		

20734	**EXTERN_FUNCTION_MASK**	
MD–Nummer	Funktionsmaske für externe Sprache	
Standardvorbesetzung:	min. Eingabegrenze: 0	max. Eingabegrenze: 16
Änderung gültig nach RESET	Schutzstufe: 2/7	Einheit: –
Datentype:DWORD	gültig ab SW–Stand: 6.2	

Bedeutung:

Mit diesem Maschinendatum werden Funktionen im ISO–Mode beeinflußt.

Bit 0 =0: ISO–Mode T: A” und “C” werden als Achsen interpretiert. Wenn Konturzug programmiert wird, muß vor “A” oder “C” ein Komma stehen.
=1: “A” und “C” im Teileprogramm werden immer als Konturzug interpretiert. Es darf keine Achse A oder C geben.

Bit 1 =0: ISO–Mode T G10 P<100 Werkzeuggeometrie
P>100 Werkzeugverschleiß
=1: G10 P<10 000 Werkzeuggeometrie
P>10 000 Werkzeugverschleiß

Bit 2 =0: G04 Verweilzeit: immer [s] oder [ms]
=1: wenn G95 aktiv ist, Verweilzeit in Spindelumdrehungen

Bit 3 =0 Fehler im ISO Scanner führen zu Alarm
Beispiel: N5 G291 ; ISO-Dialekt-Mode
N10 WAIT ; Alarm 12080 ”WAIT unbekannt”
N15 G91 G500 ; Alarm 12080 ”G500 unbekannt”
=1: Fehler im ISO Scanner werden nicht ausgegeben, es wird der Satz an den Siemenstranslator weitergegeben
Beispiel: N5 G291 ; ISO-Dialekt-Mode
N10 WAIT ; Satz wird vom Siemenstranslator
; bearbeitet
N15 G91 G500 ; Satz wird vom Siemenstranslator
; bearbeitet
N20 X Y ; Satz wird wegen G291 von ISO-Translator,
; G91 aus N15 ist aktiv

Bit 4 =0: G00 wird in der aktiven Genauhaltfunktion verfahren.
Beispiel: Bei G64 werden auch G00 Sätze mit G64 verfahren
=1 G00 Sätze werden immer mit G09 verfahren, auch wenn G64 aktiv ist

Bit 5 =0: Rundachsbewegungen werden auf dem kürzesten Weg ausgeführt
=1: Rundachsbewegungen werden abhängig vom Vorzeichen in positiver oder negativer Drehrichtung ausgeführt

Bit 6 =0: nur 4–stellige Programmnummer erlaubt
Bit 6 =1: 8–stellige Programmnummer erlaubt. Bei weniger als 4 Stellen wird mit 4 Stellen erweitert.

Bit 7 =0: Achsprogrammierung bei Geoachstausch/parallele Achsen ist ISO–Mode kompatibel
=1: Achsprogrammierung bei Geoachstausch/parallele Achsen ist im ISO–Mode kompatibel zum Siemensmode

Bit8 =0: Bei Zyklen wird der F–Wert immer als Vorschub interpretiert übergeben
=1: Bei Gewindezyklen wird der F–Wert als Steigung interpretiert übergeben

Bit9 =0: Bei ISO Mode T wird bei G84, G88 im Standardmode F bei G95 mit 0,01 mm bzw. 0,0001 inch multipliziert
=1: Bei ISO Mode T wird bei G84, G88 im Standardmode F bei G95 mit 0,01 mm bzw. 0,0001 inch multipliziert

Bit 10 = 0: Bei M96 Pxx wird beim Interrupt immer das mit Pxx progr. Programm aufgerufen.
= 1: Bei M96 Pxx wird beim Interrupt immer CYCLE396.spf aufgerufen.

Bit 11 = 0: Bei der Programmierung von G54 Pxx wird G54.1 angezeigt.
= 1: Bei der Programmierung von G54 Pxx oderG54.1 Px wird immer G54Px angezeigt.

Bit 12 = 0: Bei Aufruf des mit M96 Pxx definierten UP wird $P_ISO_STACK nicht verändert.
=1: Bei Aufruf des mit M96 Pxx definierten UP wird $P_ISO_STACK inkrementiert.

22420	**FGROUP_DEFAULT_AXES[n]: 0, ..., 7**	
MD–Nummer	Defaultwert für FGROUP–Befehl	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 8
Änderung gültig nach POWER ON	Schutzstufe: 7/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 5.2	
Bedeutung:	Man kann bis zu 8 Kanalachsen angeben, deren resultierende Geschwindigkeit dem programmierten Bahnvorschub entspricht. Stehen alle 8 Werte auf null (Vorbelegung), werden wie bisher als Defaulteinstellung für den FGROUP–Befehl die in $MC_AXCONF_GEOAX_ASSIGN_TAB eingetragenen Geo–Achsen aktiv. Beispiel: die ersten 4 Achsen im Kanal sind für den Bahnvorschub relevant: $MC_FGROUP_DEFAULT_AXES[0] = 1 $MC_FGROUP_DEFAULT_AXES[2] = 2 $MC_FGROUP_DEFAULT_AXES[3] = 3 $MC_FGROUP_DEFAULT_AXES[4] = 4	

22512	**EXTERN_GCODE_GROUPS_TO_PLC[n]: 0, ..., 7**	
MD–Nummer	Angabe der G–Gruppen, die an der Nahtstelle NCK–PLC ausgegeben werden, wennn eine externe NC–Sprache aktiv ist	
Standardvorbesetzung: –	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: BYTE	gültig ab SW–Stand: 5	
Bedeutung:	Über das Kanal–Maschinendatum $MC_EXTERN_GCODE_GROUPS_TO_PLC kann der Anwender die G–Gruppen einer externen NC–Sprache auswählen, deren aktiver G–Befehl von NCK an PLC gemeldet werden soll. Vorbelegung 0: keine Ausgabe Die Schnittstelle NCK_PLC wird mit jedem Satzwechsel und nach Reset aktualisiert. Es besteht nicht in jedem Fall ein satzsynchroner Zusammenhang, zwischen NC–Satz und gemeldeten G–Funktionen (z.B. bei kurzen Sätzen im Bahnsteuerbetrieb). Analog $MC_GCODE_GROUPS_TO_PLC	

22515	**GCODE_GROUPS_TO_PLC_MODE**	
MD–Nummer	Verhalten der G–Gruppenübergabe an PLC	
Standardvorbesetzung: –	min. Eingabegrenze: –	max. Eingabegrenze: –
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 6.3	
Bedeutung:	Zur Einstellung des Verhaltens, wie die G–Gruppen datenmäßig in der PLC zu interpretieren sind. Beim jetzigen Verhalten (Bit 0=0) ist die G–Gruppe der Array–Index eines 64 Byte großen Feldes (DBB 208 – DBB 271). Damit kann maximal die 64. G–Gruppe erreicht werden. Beim neuen Verhalten (Bit 0=1) ist die Datenablage in der PLC maximal 8 Byte (DBB 208 – DBB 215) groß. Bei diesem Verfahren ist der Array–Index dieses Byte–Arrays identisch dem Index des MD $MC_GCODE_GROUPS_TO_PLC[Index] und $MC_EXTERN_GCODE_GROUPS_TO_PLC[Index]. Hierbei darf jeder Index (0–7) nur bei einem der beiden Maschinendaten werden, beim jeweils anderen MD muß der Wert 0 eingetragen sein. Bit 0(LSB = 0: Verhalten wie bisher, das 64 Byte große Feld wird für die Anzeige der G–Codes benutzt. Bit 0(LSB = 1: Der Anwender stellt ein, für welche G–Gruppen die ersten 8 Byte benutzt werden sollen	

22900	STROKE_CHECK_INSIDE			
MD–Nummer	Richtung (innen/außen) in die der Schutzbereich wirkt			
Standardvorbesetzung: 0		min. Eingabegrenze: 0		max. Eingabegrenze: 1
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: BYTE			gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses Maschinendatum gilt in Verbindung mit externen Programmiersprachen. Es wirkt bei $MN_MM_EXTERN_LANGUAGE = 1. Es wird festgelegt, ob der Schutzbereich 3 ein Schutzbereich innen oder außen ist. Bedeutung: 0: Schutzbereich 3 ist ein Schutzbereich innen, d.h. der Schutzbereich nach innen darf nicht überfahren werden 1: Schutzbereich 3 ist ein Schutzbereich außen			

22910	WEIGHTING_FACTOR_FOR_SCALE			
MD–Nummer	Eingabefeiheit für Skalierungsfaktor			
Standardvorbesetzung: 0		min. Eingabegrenze: 0		max. Eingabegrenze: 1
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: BOOLEAN			gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses Maschinendatum gilt in Verbindung mit exernen Programmiersprachen. Es wirkt bei $MN_MM_EXTERN_LANGUAGE = 1. Festlegung der Einheit für den Skalierungsfaktor P und für die achsialen Skalierungsfaktoren I, J, K Bedeutung: 0: Skalierungsfaktor in 0.001 1: Skalierungsfaktor in 0.00001			

22914	AXES_SCALE_ENABLE			
MD–Nummer	Aktivierung für achsialen Skalierungsfaktor (G51)			
Standardvorbesetzung: 0		min. Eingabegrenze: 0		max. Eingabegrenze: 1
Änderung gültig nach POWER ON		Schutzstufe: 2/7		Einheit: –
Datentype: BOOLEAN			gültig ab SW–Stand: 5.2	
Bedeutung:	Mit diesem MD wird achsiales Skalieren freigeschaltet. Bedeutung: 0: achsiales Skalieren nicht möglich 1: achsiales Skalieren möglich, d.h. MD DEFAULT_SCALE_FACTOR_AXIS ist wirksam			

22920	**EXTERN_FIXED_FEEDRATE_F1_ON**		
SD–Nummer	Aktivierung fester Vorschübe F1 – F9		
Standardvorbesetzung: FALSE	min. Eingabegrenze:	max. Eingabegrenze:	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit:	
Datentype: BOOLEAN	gültig ab SW–Stand: 6.2		
Bedeutung:	Mit diesem MD werden die festen Vorschübe aus den Settingdaten $SC_EXTERN_FIXED_FEEDRATE_F1_F9 [ ] freigeschaltet. 0: keine festen Vorschübe mit F1 – F9 1: die Vorschübe aus den Settingdaten $SC_EXTERN_FIXED_FEEDRATE_F1_F9 werden mit der Programmierung von F1 –F9 wirksam		

22930	**EXTERN_PARALLEL_GEOAX**		
SD–Nummer	Zuordnung parallele Kanal–Geometrieachse		
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 3	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –	
Datentype: BYTE	gültig ab SW–Stand: 6.2		
Bedeutung:	Zuordnungstabelle der Achsen, die parallel zu den Geometrieachsen liegen. Über diese Tabelle können den Geometrieachsen parallel liegende Kanalachsen zugeordnet werden. Die parallelen Achsen können dann im ISO–Dialekt mit den G–Funktionen der Ebenenanwahl (G17 – G19) und dem Achsnamen der parallelen Achse als Geometrieachse aktiviert werden. Es wird dann ein Achstausch mit der über $MC_AXCONF_GEOAX_ASSIGN_TAB[ ] definierten Achse ausgeführt. Voraussetzung: Die verwendeten Kanalachsen müssen aktiv sein (belegter Listenplatz in AXCONF_MACHAX_USED). Eintrag einer Null deaktiviert die entsprechende parallele Geometrieachse.		

24004	**CHBFRAME_POWERON_MASK**		
MD–Nummer	Kanalspezifisches Basisframe nach Power On zurücksetzen		
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 0xFF	
Änderung gültig nach POWER ON	Schutzstufe: 2/7	Einheit: –	
Datentype: DWORD	gültig ab SW–Stand: 5.2		
Bedeutung:	Mit diesem Maschinendatum wird festgelegt, ob kanalspezifische Basisframes bei Power On Reset in der Datenhaltung zurückgesetzt weden. D.h. Verschiebungen und Drehungen werden auf 0, Skalierungen auf 1 gesetzt. Spiegeln wird ausgeschaltet. Die Anwahl kann für die einzelnen Basisframes getrennt erfolgen. Bit 0 entspricht Basisframe 0, Bit 1 Basisframe 1 usw. 0: Basisframe bleibt bei Power On erhalten 1: Basisframe wird bei Power in der Datenhaltung zurückgesetzt.		

6.3 Achsspezifische Settingdaten

6.3 Achsspezifische Settingdaten

43120 MD–Nummer	**DEFAULT_SCALE_FACTOR_AXIS** Achsialer Default Skalierungsfaktor bei aktivem G51	
Standardvorbesetzung: 1	min. Eingabegrenze: –99999999	max. Eingabegrenze: 99999999
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: –
Datentype: DWORD	gültig ab SW–Stand: 5.2	
Bedeutung:	Dieses Maschinendatum gilt in Verbindung mit externen Programmiersprachen. Es wirkt bei $MN_MM_EXTERN_LANGUAGE = 1. Wenn kein achsialer Skalierungsfaktor I, J oder K im G51 Satz programmiert wird, wirkt der DEFAULT_SCALEFAKTOR_AXIS. Damit der Skalierungsfaktor wirkt, muß das MD AXES_SCALE_ENABLE gesetzt sein.	

43240 MD–Nummer	**M19_SPOS** Spindelposition in Grad für Spindelpositionen mit M19	
Standardvorbesetzung: 0	min. Eingabegrenze: –359.999	max. Eingabegrenze: 359.999
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: –
Datentype: DOUBLE	gültig ab SW–Stand: 5.2	
Bedeutung:	Das Settingdatum ist auch im Siemens–Mode wirksam.	

6.4 Kanalspezifische Settingdaten

42110 SD–Nummer	**DEFAULT_FEED** Defaultwert für Bahnvorschub	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: –
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: –
Datentype: DOUBLE	gültig ab SW–Stand: 5.2	
Bedeutung:	Ist im Teileprogramm kein Bahnvorschub programmiert, so wird der in $SC_DE-FAULT_FEED abgelegte Wert verwendet. Die Auswertung des Settingdatums erfolgt beim Teileprogrammstart unter Berücksichtigung des zu diesem Zeitpunkt wirksamen Vorschubtyps (siehe $MC_GCODE_RESET_VALUES bzw. $MC_EXTERN_GCODE_RESET_VALUES).	

42140 SD–Nummer	**DEFAULT_SCALE_FACTOR_P** Default Skalierungsfaktor für Adresse P	
Standardvorbesetzung: 0	min. Eingabegrenze: –99999999	max. Eingabegrenze: 99999999
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: –

42140	DEFAULT_SCALE_FACTOR_P
SD–Nummer	Default Skalierungsfaktor für Adresse P
Datentype: DWORD	gültig ab SW–Stand: 5.2
Bedeutung:	Dieses Maschinendatum gilt in Verbindung mit externen Programmiersprachen. Es wirkt bei $MN_MM_EXTERN_LANGUAGE = 1. Wenn kein Skalierungsfaktor P im Satz programmiert ist wirkt der Wert aus diesem Maschinendatum.

42150	DEFAULT_ROT_FACTOR_R	
SD–Nummer	Vorbelegung für Rotationswinkel R	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: 360
Änderung gültig SOFORT	Schutzstufe: 2/7	Einheit: grad
Datentype: DOUBLE	gültig ab SW–Stand:	
Bedeutung:	Wenn kein Faktor für Rotation R bei der Anwahl der Rotation G68 porgrammiert ist, wirkt der Wert aus diesem Settingdatum.	

42160	EXTERN_FIXED_FEEDRATE_F1_F9	
SD–Nummer	Feste Vorschübe mit F1 – F9	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze:
Änderung gültig SOFORT	Schutzstufe: 2/7	Einheit: VELO
Datentype: DOUBLE	gültig ab SW–Stand:	
Bedeutung:	Feste Vorschübe, die mit der Programmierung von F1 – F9 und aktivem G01 angewählt werden.	

42162	EXTERN_DOUBLE_TURRET_DIST	
SD–Nummer	Werkzeugabstand des Doppelrevolverkopfs	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig	Schutzstufe:	Einheit:
Datentype: DOUBLE	gültig ab SW–Stand:	
Bedeutung:	Das Maschinendatum ist nur wirksam bei $MN_EXTER_CNC_SYSTEM = 2. Turning Abtand der beiden Werkzeuge eines Doppelschlittenrevolverkopfs. Der Abstand wird mit G68 als additive Nullpunktverschiebung aktiviert, wenn $MN_EXTERN_DOUBLE_TURRET_ON = TRUE gesetzt ist.	

42520	CORNER_SLOWDOWWN_START	
SD–Nummer	Beginn der Vorschubreduzierung bei G62	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: beliebig
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: POSN_LIN

42520	**CORNER_SLOWDOWWN_START**	
SD–Nummer	Beginn der Vorschubreduzierung bei G62	
Datentype: DOUBLE	gültig ab SW–Stand: 6	
Bedeutung:	Bahnweglänge, ab der der Vorschub vor der Ecke bei G62 reduziert wird	

42522	**CORNER_SLOWDOWN_END**	
SD–Nummer	Ende der Vorschubreduzierung bei G62	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: beliebig
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: POSN_LIN
Datentype: DOUBLE	gültig ab SW–Stand: 6	
Bedeutung:	Bahnweglänge, bis zu der der Vorschub nach einer Ecke bei G62 reduziert bleibt.	

42524	**CORNER_SLOWDOWN_OVR**	
SD–Nummer	Override zur Vorschubreduzierung bei G62	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: beliebig
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: PERCENT
Datentype: DOUBLE	gültig ab SW–Stand: 6	
Bedeutung:	Override, mit dem der Vorschub an der Ecke bei G62 multipliziert wird.	

42526	**CORNER_SLOWDOWN_CRIT**	
SD–Nummer	Eckenerkennung bei G62, G21	
Standardvorbesetzung: 0	min. Eingabegrenze: 0	max. Eingabegrenze: beliebig
Änderung gültig SOFORT	Schutzstufe: 7/7	Einheit: POSN_ROT
Datentype: DOUBLE	gültig ab SW–Stand: 6	
Bedeutung:	Winkel, ab dem eine Ecke bei der Vorschubreduzierung mit G62, G21 berücksichtigt wird.	

43340	**EXTERN_REF_POSITION_G30_1**	
SD–Nummer	Referenzpunktposition für G30.1	
Standardvorbesetzung:	min. Eingabegrenze:	max. Eingabegrenze:
Änderung gültig SOFORT	Schutzstufe:	Einheit:
Datentype: DOUBLE	gültig ab SW–Stand:	
Bedeutung:	Settingdatum Referenzpunktposition für G30.1. Dieses Settingdatum wird im CYCLE328 ausgewertet.	

■

7 Signalbeschreibungen

keine

8 Beispiel

keine

Platz für Notizen

9 Datenfelder, Listen

9.1 Maschinendaten

Nummer	Bezeichner	Name	Verweis
allgemeine ($MN_ ...)			
10604	WALIM_GEOAX_CHANGE_MODE	Arbeitsfeldbegrenzung beim Umschalten von Geoachsen	
10615	NCFRAME_POWERON_MASK	Globale Basisframes nach Power On löschen	K2
10652	CONTOUR_DEF_ANGLE_NAME	Einstellbarer Name für Winkel in der Kontur–Kurzbeschreibung	
10654	RADIUS_NAME	Einstellbarer Name für Radius satzweise in der Kontur–Kurzbeschreibung	
10656	CHAMFER_NAME	Einstellbarer Name für Fase in der Kontur–Kurzbeschreibung	
10704	DRYRUN_MASK	Aktivierung des Probelaufvorschubs	
10706	SLASH_MASK	Aktivierung der Satzausblendung	
10715	M_NO_FCT_CYCLE[n]: 0, ..., 0	M–Funktionsnummer für Werkzeugwechselzyklenaufruf	K1
10716	M_NO_FCT_CYCLE_NAME[ ]	Name für Werkzeugwechselzyklus bei M–Funktionen aus MD $MN_MFCT_CYCLE	K1
10717	T_NO_FCT_CYCLE_NAME	Name für Werkzeugwechselzyklus für T–Funktion	K1
10718	M_NO_FCT_CYCLE_PAR	M–Funktionsersetzung mit Parametern	K1
10719	T_NO_FCT_CYCLE_MODE	Parametrierung der T–Funktions–Ersetzung	K1
10740	EXTER_M_NO_MAC_CYCLE	Makroaufruf über M-Funktion	
10741	EXTER_M_NO_MAC_CYCLE_NAME	UP-Name für M-Funktion Makroaufruf	
10760	G53_TOOLCORR	Wirkungsweise G53, G153 und SUPA	
10800	EXTERN_CHAN_SYNC_M_NO_MIN	Erste M–Nummer für Kanalsynchronisation	
10802	EXTERN_CHAN_SYNC_M_NO_MAX	Letzte M–Nummer für Kanalsynchronisation	
10804	EXTERN_M_NO_SET_INT	M–Funktion für ASUP–Aktivierung	
10806	EXTERN_M_NO_DISABLE_INT	M–Funktion für ASUP–Deaktivierung	
10808	EXTERN_INTERRUPT_BITS_M96	Interruptprogramm–Bearbeitung (M96)	
10810	EXTERN_MEAS_G31_P_SIGNAL	Zuordnung der Meßeingänge für G31 P..	
10812	EXTERN_DOUBLE_TURRET_ON	Doppelrevolverkopf mit G68	
10814	EXTERN_M_NO_MAC_CYCLE	Makroaufruf über M–Funktion	
10815	EXTERN_M_NO_MAC_CYCLE_NAME	UP–Name für M–Funktion Makroaufruf	

allgemeine ($MN_ ...)			
10816	EXTERN_G_NO_MAC_CYCLE	IMakroaufruf über G–Funktion	
10817	EXTERN_G_NO_MAC_CYCLE_NAME	UP–Name für G–Funktion Makroaufruf	
10818	EXTERN_INTERRUPT_NUM_ASUP	Interruptnummer für ASUP–Start (M96)	
10820	EXTERN_INTERRUPT_NUM_RETRAC	Interruptnummer für Schnellrückzug (G10.6)	
10870	EXTERN_CHAN_SYNC_M_NO_MIN	Erste M-Nummer für Kanalsynchronisation	
10872	EXTERN_CHAN_SYNC_M_NO_MAX	Letzte M-Nummer für Kanalsynchronisation	
10880	EXTERN_CNC_SYSTEM	Externes Steuerungssystem, dessen Programme abgearbeitet werden sollen	
10881	EXTERN_GCODE_SYSTEM	ISO Mode T: G Code System	
10882	NC_USER_EXTERN_GCODES_TAB[n]: 0–59	Liste anwenderspezifischer G–Befehle einer externen NC–Sprache	
10884	EXTERN_FLOATINGPOINT_PROG	Bewertung programmierter Werte ohne Dezimalpunkt	
10886	EXTERN_INCREMENT_SYSTEM	Festlegung des Inkrement–Systems	
10888	EXTERN_DIGITS_TOOL_NO	Stellenanzahl für T–Nummer in externem Sprachmode	
10890	EXTERN_TOOLPROG_MODE	Werkzeugwechsel–Programmierung bei externer Programmiersprache	
18190	MM_NUM_PROTECT_AREA_NCK	Anzahl der Dateien für maschinenbezogene Schutzbereiche (SRAM)	S7
18800	MM_EXTERN_LANGUAGE	Aktivierung externer NC–Sprachen	
kanalspezifisch ($MC_ ...)			
20050	AXCONF_GEOAX_ASSIGN_TAB[ ]	Zuordnung Geometrieachse zu Kanalachse	K2
20060	AXCONF_GEOAX_NAME_TAB[ ]	Geometrieachse im Kanal	K2
20070	AXCONF_MACHAX_USED[ ]	Maschinenachsnummer gültig im Kanal	K2
20080	AXCONF_CHANAX_NAME_TAB[ ]	Kanalachsname im Kanal	K2
20094	SPIND_RIGID_TAPPING_M_NR	M–Funktion für das Umschalten in den gesteuerten Achsbetrieb	
20095	EXTERN_RIGID_TAPPING_M_NR	M–Funktionsnummer im externer Sprachmodus für die Umschaltung der Spindel in den gesteuerten Spindelbetrieb	
20100	DIAMETER_AX_DEF	Geometrieachse mit Planachsfunktion	P1
20150	GCODE_RESET_VALUES[n]: 0 bis max. Anzahl G–Codes	Löschstellung der G–Gruppen	K1
20152	GCODE_RESET_MODE	Resetverhalten der G-Gruppe	
20154	EXTERN_GCODE_RESET_VALUES[n]: 0–30	Löschstellung der G–Gruppen	
20380	TOOL_CORR_MODE_G43G44	Behandlung der Werkzeuglängenkorrektur G43/G44	
20382	TOOL_CORR_MOVE_MODE	Herausfahren der Werkzeuglängenkorrektur	
20732	EXTERN_G0_LINEAR_MODE	Interpolationsverhalten bei G00 festlegen	
20734	EXTERN_FUNCTION_MASK	Funktionsmaske für externe Sprache	
22420	FGROUP_DEFAULT_AXES[ ]	Defaultwert für FGROUP–Befehl	
22512	EXTERN_GCODE_GROUPS_TO_PLC[n]: 0–7	Sende G–Befehl einer externen NC–Sprache an PLC	

kanalspezifisch ($MC_ ...)			
22515	GCODE_GROUPS_TO_PLC_MODE	Verhalten der G–Gruppenübergabe an PLC	
22900	STROKE_CHECK_INSIDE	Richtung (innen/außen) in die der Schutzbereich wirkt	
22910	WEIGHTING_FACTOR_FOR_SCALE	Einheit des Skalierungsfaktors	
22914	AXES_SCALE_ENABLE	Aktivierung für achsialen Skalierungsfaktor (G51)	
22920	EXTERN_FEEDRATE_F1_F9_ACTIV	Feste Vorschübe mit F0 – F9 erlauben	
22930	EXTERN_PARALLEL_GEOAX	Zuordnung parallele Kanal–Geometrieachse	
24004	CHBFRAME_POWERON_MASK	Kanalspezifisches Basisframe nach Power On zurücksetzen	
28080	NUM_USER_FRAMES	Anzahl der Nullpunktverschiebungen	
29210	NUM_PROTECT_AREA_ACTIVE	Schutzbereich aktivieren	
34100	REFP_SET_POS[0]	Referenzpunktwert / bei abstandkodiertem System ohne Bedeutung	
35000	SPIND_ASSIGN_TO_MACHAX	Zuordnung Spindel zu Maschinenachse	

9.2 Settingdaten

Nummer	Bezeichner	Name	Verweis
achsspezifisch			
42150	DEFAULT_ROT_FACTOR_R	Vorbelegung für Rotationswinkel R	
43120	DEFAULT_SCALE_FACTOR_AXIS	Achsialer Default Skalierungsfaktor bei aktivem G51	
43240	M19_SPOS	Position der Spindel bei Programmierung von M19	
42890	M19_SPOSMODE	Positioniermodus der Spindel bei Programmierung von M19	
kanalspezifisch			
42110	DEFAULT_FEED	Defaultwert für Bahnvorschub	V1
42140	DEFAULT_SCALE_FACTOR_P	Default Skalierungsfaktor für Adresse P	
42150	DEFAULT_ROT_FACTOR_R	Vorbelegung für Rotationswinkel R	
42160	EXTERN_FIXED_FEEDRATE_F1_F9	Feste Vorschübe mit F1 – F9	
42162	EXTERN_DOUBLE_TURRET_DIST	Werkzeugabstand des Doppelrevolverkopfes	
42520	CORNER_SLOWDOWN_START	Beginn der Vorschubreduzierung bei G62	
42522	CORNER_SLOWDOWN_END	Ende der Vorschubreduzierung bei G62	
42524	CORNER_SLOWDOWN_OVR	Override zur Vorschubreduzierung bei G62	

kanalspezifisch			
42526	CORNER_SLOWDOWN_CRIT	Eckenerkennung bei G62, G21	
43340	EXTERN_REF_POSITION_G30_1	Referenzpunktposition für G30.1	

■

10 Alarme

Werden in Zyklen fehlerhafte Zustände erkannt, so wird ein Alarm erzeugt und die Abarbeitung des Zyklus unterbrochen.

Weiterhin geben die Zyklen Meldungen in der Dialogzeile der Steuerung aus. Diese Meldungen unterbrechen die Bearbeitung nicht.

In den Zyklen werden Alarme mit Nummern zwischen 61000 und 62999 generiert (siehe /DA/, Diagnoseanleitung und /PGZ/, Zyklen, Programmieranleitung). Dieser Nummernbereich ist hinsichtlich der Alarmreaktionen und Löschkritierien nochmals unterteilt.

In der nachstehenden Tabelle finden Sie außerdem Fehlermeldungen bei den in Kapitel 3 beschriebenen Zyklen.

Tabelle 10-1 Alarmnummer und Alarmbeschreibung

Alarm–Nr.	Kurzbeschreibung	Quelle	Erläuterung/Abhilfe
Alarme – allgemein			
61001	Gewindesteigung falsch definiert	CYCLE376T	Gewindesteigung falsch definiert
61003	Kein Vorschub im Zyklus programmiert	CYCLE371T, CYCLE374T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Im Aufrufsatz oder vor dem Zyklusaufruf wurde kein Vorschub “F” programmiert, siehe Siemens–Standardzyklen
61004	Konfiguration Geometrieachse nicht korrekt	CYCLE328	Die Reihenfolge der Geometrieachsen ist falsch, siehe Siemens–Standardzyklen
61101	Referenzebene falsch definiert	CYCLE375T, CYCLE81, CYCLE83, CYCLE84, CYCLE87	Siehe Siemens–Standardzyklen
61102	Keine Spindelrichtung programmiert	CYCLE371T, CYCLE374T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Spindelrichtung M03 oder M04 fehlt, siehe Siemens–Standardzyklen
61107	Erste Bohrtiefe falsch definiert		Erste Bohrtiefe liegt entgegengesetzt zur Gesamtbohrtiefe
61603	Einstechform falsch definiert	CYCLE374T	Einstichtiefe hat den Wert 0
61607	Startpunkt falsch definiert	CYCLE376T	Der vor Zyklenaufruf erreichte Startpunkt liegt nicht außerhalb des zu bearbeitenden Bereiches
61610	Keine Zustelltiefe programmiert	CYCLE374T	Zustelltiefe hat den Wert 0

Tabelle 10-1 Alarmnummer und Alarmbeschreibung

Alarm–Nr.	Kurzbeschreibung	Quelle	Erläuterung/Abhilfe
ISO–Alarme			
61800	Externes CNC–System fehlt	CYCLE300, CYCLE328, CYCLE330, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Maschinendatum für externe Sprache MD18800: $MN_MM_EXTERN_LANGUAGE bzw. Optionsbit 19800 $ON_EXTERN_LANGUAGE ist nicht gesetzt
61801	Falscher G–Code angewählt	CYCLE300, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T	Im Programmaufruf CYCLE300<Wert> wurde ein, für das eingegebene CNC–System, unzulässiger Zahlenwert programmiert oder in dem Zyklen–Setting–Datum wurde ein falscher Wert für das G–Code–System gegeben.
61802	Falscher Achstyp	CYCLE328, CYCLE330	Die programmierte Achse ist einer Spindel zugeordnet
61803	Programmierte Achse nicht vorhanden	CYCLE328, CYCLE330	Die programmierte Achse ist im System nicht vorhanden. MD20050–20080 prüfen
61804	Programmierte Position überschreitet Referenzpunkt	CYCLE328, CYCLE330	Die programmierte Zwichenposition oder aktuelle Position befindet sich hinter dem Referenzpunkt
61805	Wert absolut und inkremental programmiert	CYCLE328, CYCLE330, CYCLE371T, CYCLE374T, CYCLE376T, CYCLE383T, CYCLE384T, CYCLE385T	Die programmierte Zwischenposition ist sowohl absolut als auch inkremental programmiert
61806	Falsche Achszuordnung	CYCLE328	Die Reihenfolge der Achszuordnung ist falsch
61807	Falsche Spindelrichtung programmiert (aktiv)	CYCLE384M	Die programmierte Spindelrichtung widerspricht der für den Zyklus vorgesehenen Spindelrichtung
61808	Endbohrtiefe oder Einzelbohrtiefe fehlt	CYCLE383T, CYCLE384T, CYCLE385T, CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	Gesamttiefe “Z” oder Einzelbohrtiefe “Q” fehlt im G8x–Satz (Erstaufruf des Zyklus)
61809	Bohrposition nicht zulässig	CYCLE383T, CYCLE384T, CYCLE385T	
61810	ISO–GCode nicht möglich	CYCLE383T, CYCLE384T, CYCLE385T	
61811	ISO–Achsname nicht zulässig	CYCLE328, CYCLE330. CYCLE371T, CYCLE374T, CYCLE376T, CLE383T, CYCLE384T, CYCLE385T	Im Aufrufsatz wurde ein nicht zulässiger ISO–Achsname programmiert
61812	Wert(e) im externen Zyklusaufruf falsch definiert	CYCLE371T, CYCLE376T,	Im Aufrufsatz wurde ein nicht zulässiger Zahlenwert programmiert
61813	GUD–Wert falsch definiert	CYCLE376T	In den Zyklen–Settingdaten wurde ein unzulässiger Zahlenwert eingegeben

Tabelle 10-1 Alarmnummer und Alarmbeschreibung

Alarm–Nr.	Kurzbeschreibung	Quelle	Erläuterung/Abhilfe
61814	Polarkoordinaten nicht möglich	CYCLE381M, CYCLE383M, CYCLE384M, CYCLE387M	
61815	G40 nicht aktiv	CYCLE374T, CYCLE376T	Vor dem Zyklusaufruf war G40 nicht aktiv
61816	Achsen nicht auf Referenzpunkt		
61817	Achskoordinaten innerhalb des Schutzbereiches		
61818	Achsbereichsgrenzwerte sind gleich		

■

Platz für Notizen

Literatur

A

Allgemeine Dokumentation

/BU/

SINUMERIK & SIMODRIVE, Automatisierungssysteme für Bearbeitungsmaschinen
Katalog NC 60
Bestellnummer: E86060–K4460–A101–B1
Bestellnummer: E86060–K4460–A101–B1–7600 (englisch)

/IKPI/

Industrielle Kommunikation und Feldgeräte
Katalog IK PI
Bestellnummer: E86060–K6710–A101–B2
Bestellnummer: E86060–K6710–A101–B2–7600 (englisch)

/ST7/

SIMATIC
Produkte für Totally Integrated Automation und Micro Automation
Katalog ST 70
Bestellnummer: E86060–K4670–A111–A8
Bestellnummer: E86060–K4670–A111–A8–7600 (englisch)

/Z/

MOTION–CONNECT
Verbindungstechnik & Systemkomponenten für SIMATIC, SINUMERIK, MASTERDRIVES und SIMOTION
Katalog NC Z
Bestellnummer: E86060–K4490–A001–B1
Bestellnummer: E86060–K4490–A001–B1–7600 (englisch)

Safety Integrated Applikationshandbuch
Das Sicherheitsprogramm für die Industrien der Welt
Bestellnummer: 6ZB5000–0AA01–0BA0
Bestellnummer: 6ZB5000–0AA02–0BA0 (englisch)

Elektronische Dokumentation

/CD1/

Das SINUMERIK–System (Ausgabe 03. 04)
DOC ON CD
(mit allen SINUMERIK 840D/840Di/810D/802– und SIMODRIVE– Schriften)
Bestellnummer: 6FC5298–7CA00–0AG0

Anwender–Dokumentation

/AUK/ SINUMERIK 840D/810D (Ausgabe 09.99)
Kurzanleitung **Bedienung AutoTurn**
Bestellnummer: 6FC5298–4AA30–0AP2

/AUP/ SINUMERIK 840D/810D (Ausgabe 02.02)
Bedienungsanleitung **Grafisches Programmiersystem AutoTurn**
Programmieren / Einrichten
Bestellnummer: 6FC5298–4AA40–0AP3

/BA/ SINUMERIK 840D/810D (Ausgabe 10.00)
Bedienungsanleitung **MMC**
Bestellnummer: 6FC5298–6AA00–0AP0

/BAD/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Bedienungsanleitung **HMI Advanced**
Bestellnummer: 6FC5298–6AF00–0AP3

/BAH/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Bedienungsanleitung **HT 6**
Bestellnummer: 6FC5298–0AD60–0AP3

/BAK/ SINUMERIK 840D/840Di/810D (Ausgabe 02.01)
Kurzanleitung Bedienung
Bestellnummer: 6FC5298–6AA10–0AP0

/BAM/ SINUMERIK 810D/840D (Ausgabe 08.02)
Bedienen/Programmieren **ManualTurn**
Bestellnummer: 6FC5298–6AD00–0AP0

/BAS/ SINUMERIK 840D/840Di/810D (Ausgabe 11.03)
Bedienen/Programmieren **ShopMill**
Bestellnummer: 6FC5298–6AD10–0AP2

/BAT/ SINUMERIK 840D/810D (Ausgabe 06.03)
Bedienen/Programmieren **ShopTurn**
Bestellnummer: 6FC5298–6AD50–0AP2

/BEM/ SINUMERIK 840D/810D (Ausgabe 03.04)
Bedienungsanleitung **HMI Embedded**
Bestellnummer: 6FC5298–6AC00–0AP3

/BNM/ SINUMERIK 840D840Di//810D (Ausgabe 03.04)
Benutzerhandbuch **Messzyklen**
Bestellnummer: 6FC5298–6AA70–0AP3

/BTDI/ SINUMERIK 840D840Di//810D (Ausgabe 06.04)
Motion Control Information System (MCIS)
Benutzerhandbuch **Tool Data Information**
Bestellnummer: 6FC5297–6AE01–0AP1

/CAD/ SINUMERIK 840D/840Di/810D (Ausgabe 03.02)
Bedienungsanleitung **CAD–Reader**
Bestellnummer: (ist Bestandteil der Online–Hilfe)

/DA/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Diagnoseanleitung
Bestellnummer: 6FC5298–7AA20–0AP0

/KAM/ SINUMERIK 840D/810D (Ausgabe 04.01)
Kurzanleitung **ManualTurn**
Bestellnummer: 6FC5298–5AD40–0AP0

/KAS/ SINUMERIK 840D/810D (Ausgabe 04.01)
Kurzanleitung **ShopMill**
Bestellnummer: 6FC5298–5AD30–0AP0

/KAT/ SINUMERIK 840D/810D (Ausgabe 07.01)
Kurzanleitung **ShopTurn**
Bestellnummer: 6FC5298–6AF20–0AP0

/PG/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Programmieranleitung **Grundlagen**
Bestellnummer: 6FC5298–7AB00–0AP0

/PGA/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Programmieranleitung **Arbeitsvorbereitung**
Bestellnummer: 6FC5298–7AB10–0AP0

/PGA1/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Listenhandbuch **Systemvariablen**
Bestellnummer: 6FC5298–7AE10–0AP0

/PGK/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Kurzanleitung **Programmierung**
Bestellnummer: 6FC5298–7AB30–0AP0

/PGM/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Programming Guide **ISO Milling**
Bestellnummer: 6FC5298–6AC20–0BP2

/PGT/ SINUMERIK 840D/840Di/810D (11.02 Edition)
Programming Guide **ISO Turning**
Bestellnummer: 6FC5298–6AC10–0BP2

/PGZ/ SINUMERIK 840D840Di//810D (Ausgabe 03.04)
Programmieranleitung **Zyklen**
Bestellnummer: 6FC5298–7AB40–0AP0

/PI / PCIN 4.4
Software zur Datenübertragung an/von **MMC–Modul**
Bestellnummer: 6FX2060–4AA00–4XB0 (dt., engl., frz.)
Bestellort: WK Fürth

/SYI/ SINUMERIK 840Di (Ausgabe 02.01)
Systemüberblick
Bestellnummer: 6FC5298–6AE40–0AP0

Hersteller–/Service–Dokumentation

a) Listen

/LIS/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
SIMODRIVE 611D
Listen
Bestellnummer: 6FC5297–7AB70–0AP0

b) Hardware

/ASAL/ SIMODRIVE 611, MASTERDRIVES VC/MC (Ausgabe 10.03)
Projektierungsanleitung Allgemeiner Teil für **Asynchron–Servomotoren**
Bestellnummer: 6SN1197–0AC62–0AP0

/APH2/ SIMODRIVE 611 (Ausgabe 10.03)
Projektierungsanleitung **Asynchron–Servomotoren1PH2**
Bestellnummer: 6SN1197–0AC63–0AP0

/APH4/ SIMODRIVE 611 (Ausgabe 10.03)
Projektierungsanleitung **Asynchron–Servomotoren 1PH4**
Bestellnummer: 6SN1197–0AC64–0AP0

/APH7M/ MASTERDRIVES MC (Ausgabe 04.04)
Projektierungsanleitung **Asynchron–Servomotoren 1PH7**
Bestellnummer: 6SN1197–0AC66–0AP0

/APH7S/ SIMODRIVE 611 (Ausgabe 01.04)
Projektierungsanleitung **Asynchron–Servomotoren 1PH7**
Bestellnummer: 6SN1197–0AC65–0AP0

/APL6/ MASTERDRIVES VC/MC (Ausgabe 01.04)
Projektierungsanleitung **Asynchron–Servomotoren 1PL6**
Bestellnummer: 6SN1197–0AC67–0AP0

/BH/ SINUMERIK 840D840Di//810D (Ausgabe 11.03)
Handbuch **Bedienkomponenten**
Bestellnummer: 6FC5297–6AA50–0AP3

/BHA/ SIMODRIVE Sensor (Ausgabe 03.03)
Benutzerhandbuch (HW) **Absolutwertgeber mit Profibus–DP**
Bestellnummer: 6SN1197–0AB10–0YP2

/EMV/ SINUMERIK, SIROTEC, SIMODRIVE (Ausgabe 06.99)
Projektierungsanleitung **EMV–Aufbaurichtlinie**
Bestellnummer: 6FC5297–0AD30–0AP1

Die aktuelle Konformitätserklärung finden Sie im Internet unter
http://www4.ad.siemens.de

Bitte geben Sie dort die ID–Nr.: 15257461 in das Feld ”Suche” ein (rechts oben) und klicken Sie auf ”go”.

/GHA/ SINUMERIK/ SIMOTION (Ausgabe 02.03)
ADI4 – Analoge Antriebsschnittstelle für 4 Achsen
Gerätehandbuch
Bestellnummer: 6FC5297–0BA01–0AP1

/PFK6/ SIMODRIVE 611, MASTERDRIVES MC (Ausgabe 05.03)
Projektierungsanleitung **Synchron–Servomotoren 1FK6**
Bestellnummer: 6SN1197–0AD05–0AP0

/PFK7/ SIMODRIVE 611, MASTERDRIVES MC (Ausgabe 01.03)
Projektierungsanleitung **Synchron–Servomotoren 1FK7**
Bestellnummer: 6SN1197–0AD06–0AP0

/PFS6/ MASTERDRIVES MC (Ausgabe 04.04)
Projektierungsanleitung **Synchron– Servomotoren 1FS6**
Bestellnummer: 6SN1197–0AD08–0AP1

/PFT5/ SIMODRIVE (Ausgabe 05.03)
Projektierungsanleitung **Synchron–Servomotoren 1FT5**
Bestellnummer: 6SN1197–0AD01–0AP0

/PFT6/ SIMODRIVE 611, MASTERDRIVES MC (Ausgabe 01.04)
Projektierungsanleitung **Synchron–Servomotoren 1FT6**
Bestellnummer: 6SN1197–0AD02–0AP0

/PFU/ SINAMICS, MASTERDRIVES (Ausgabe 09.03)
MICROMASTER
SIEMOSYN–Motoren 1FU8
Bestellnummer: 6SN1197–0AC80–0AP0

/PHC/ SINUMERIK 810D (Ausgabe 11.02)
Handbuch **Projektierung (HW)**
Bestellnummer: 6FC5297–6AD10–0AP1

/PHD/ SINUMERIK 840D (Ausgabe 11.03)
Handbuch **Projektierung (HW)**
Bestellnummer: 6FC5297–6AC10–0AP3

/PJAL/
SIMODRIVE 611, SIMOVERT MASTERDRIVES MC (Ausgabe 01.04)
Projektierungsanleitung **Synchron–Servomotoren**
Allgemeiner Teil für 1FT– / 1FK–Motoren
Bestellnummer: 6SN1197–0AD07–0AP1

/PJAS/
SIMODRIVE 611, MASTERDRIVES VC/MC (Ausgabe 06.04)
Projektierungsanleitung für Asynchronmotoren
Inhalt: Allgemeiner Teil, 1PH2, 1PH4, 1PH7, 1PL6
Bestellnummer: 6SN1197–0AC61–0AP0

/PJFE/
SIMODRIVE (Ausgabe 03.04)
Projektierungsanleitung **Synchron–Einbaumotoren 1FE1**
Drehstrommotoren für Hauptspindelantriebe
Bestellnummer: 6SN1197–0AC00–0AP5

/PJF1/
SIMODRIVE (Ausgabe 12.02)
Montageanleitung **Synchron–Einbaumotoren 1FE1 051.–1FE1 147**.
Drehstrommotoren für Hauptspindelantriebe
Bestellnummer: 610.43000.02

/PJLM/
SIMODRIVE (Ausgabe 06.02)
Projektierungsanleitung **Linearmotoren 1FN1, 1FN3**
ALL Allgemeines zum Linearmotor
1FN1 Drehstrom Linearmotor 1FN1
1FN3 Drehstrom Linearmotor 1FN3
CON Anschlußtechnik
Bestellnummer: 6SN1197–0AB70–0AP4

/PJM2/
SIMODRIVE 611, MASTERDRIVES MC (Ausgabe 06.04)
Projektierungsanleitung für Synchron–Servomotoren
Inhalt: Allgemeiner Teil, 1FT5, 1FT6, 1FK6, 1FK7, 1FS6
Bestellnummer: 6SN1197–0AC20–0AP0

/PJTM/
SIMODRIVE (Ausgabe 05.03)
Projektierungsanleitung **Einbau–Torquemotoren 1FW6**
Bestellnummer: 6SN1197–0AD00–0AP1

/PJU/
SIMODRIVE 611 (Ausgabe 02.03)
Projektierungsanleitung **Umrichter**
Bestellnummer: 6SN1197–0AA00–0AP6

/PKTM/
SIMOVERT MASTERDRIVES (Ausgabe 03.04)
Projektierungsanleitung **Komplett–Torquemotoren 1FW3**
Bestellnummer: 6SN1197–0AC70–0AP0

/PMH/
SIMODRIVE Sensor (Ausgabe 07.02)
Projektierungs–/Montageanleitung
Hohlwellenmesssystem SIMAG H
Bestellnummer: 6SN1197–0AB30–0AP1

/PMH2/ SIMODRIVE Sensor (Ausgabe 03.04)
Projektierungs–/Montageanleitung
Hohlwellenmesssystem SIMAG H2
Bestellnummer: 6SN1197–0AB31–0AP0

/PMHS/ SIMODRIVE (Ausgabe 12.00)
Montageanleitung **Messsystem für Hauptspindelantriebe**
Zahnradgeber SIZAG2
Bestellnummer: 6SN1197–0AB00–0YP3

/PMS/ SIMODRIVE (Ausgabe 03.04)
Projektierungsanleitung
ECO–Motorspindel für Hauptspindelantriebe 2SP1
Bestellnummer: 6SN1197–0AD04–0AP1

/PPH/ SIMODRIVE (Ausgabe 12.01)
Projektierungsanleitung **1PH2–/1PH4–/1PH7–Motoren**
Drehstrom–Asynchronmotoren für Hauptspindelantriebe
Bestellnummer: 6SN1197–0AC60–0AP0

/PPM/ SIMODRIVE (Ausgabe 11.01)
Projektierungsanleitung
Hohlwellenmotoren für Hauptspindelantriebe **1PM4 und 1PM6**
Bestellnummer: 6SN1197–0AD03–0AP0

c) Software

/FB1/ SINUMERIK 840D/840Di/810D/FM–NC (Ausgabe 03.04)
Funktionsbeschreibung **Grundmaschine (Teil 1)**
(im folgenden sind die enthaltenen Bücher aufgeführt)
Bestellnummer: 6FC5297–7AC20–0AP0
A2 Diverse Nahtstellensignale
A3 Achsüberwachungen, Schutzbereiche
B1 Bahnsteuerbetrieb, Genauhalt und Look Ahead
B2 Beschleunigung
D1 Diagnosehilfsmittel
D2 Dialogprogrammierung
F1 Fahren auf Festanschlag
G2 Geschwindigkeiten, Soll–/Istwertsysteme, Regelung
H2 Hilfsfunktionsausgabe an PLC
K1 BAG, Kanal, Programmbetrieb
K2 Achsen, Koordinatensysteme, Frames,
Werkstücknahes Istwertsystem, Externe Nullpunktverschiebung
K4 Kommunikation
N2 NOT AUS
P1 Planachsen
P3 PLC–Grundprogramm
R1 Referenzpunktfahren
S1 Spindeln
V1 Vorschübe
W1 Werkzeugkorrektur

/FB2/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Funktionsbeschreibung **Erweiterungsfunktionen (Teil 2)**
einschließlich FM–NC: Drehen, Schrittmotor
(im folgenden sind die enthaltenen Bücher aufgeführt)
Bestellnummer: 6FC5297–7AC30–0AP0
A4 Digitale und analoge NCK–Peripherie
B3 Mehrere Bedientafeln und NCUs
B4 Bedienung über PG/PC
F3 Ferndiagnose
H1 Handfahren und Handradfahren
K3 Kompensationen
K5 BAGs, Kanäle, Achstausch
L1 FM–NC lokaler Bus
M1 Kinematische Transformation
M5 Messen
N3 Softwarenocken, Wegschaltsignale
N4 Stanzen und Nibbeln
P2 Positionierachsen
P5 Pendeln
R2 Rundachsen
S3 Synchronspindel
S5 Synchronaktionen (bis SW 3 / danach /FBSY/)
S6 Schrittmotorsteuerung
S7 Speicherkonfiguration
T1 Teilungsachsen
W3 Werkzeugwechsel
W4 Schleifen

/FB3/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Funktionsbeschreibung **Sonderfunktionen (Teil 3)**
(im folgenden sind die enthaltenen Bücher aufgeführt)
Bestellnummer: 6FC5297–7AC80–0AP0
F2 3– bis 5–Achs–Transformation
G1 Gantry–Achsen
G3 Taktzeiten
K6 Konturtunnelüberwachung
M3 Achskopplung und ESR
S8 Konstante Werkstückdrehzahl für Centerless Schleifen
S9 Sollwertschaltung (S9)
T3 Tangentialsteuerung
TE0 Installation und Aktivierung der Compilezyklen
TE1 Abstandsregelung
TE2 Analoge Achse
TE3 Drehzahl–/ Drehmomentkopplung Master–Slave
TE4 Transformationspacket Handling
TE5 Sollwertumschaltung
TE6 MKS–Kopplung
TE7 Wiederaufsetzen – Retrace Support
TE8 Taktunabhängige bahnsynchrone Schaltsignalausgabe
V2 Vorverarbeitung
W5 3D–Werkzeugradiuskorrektur

/FBA/ SIMODRIVE 611D/SINUMERIK 840D/810D (Ausgabe 03.04)
Funktionsbeschreibung **Antriebsfunktionen**
(im folgenden sind die enthaltenen Kapitel aufgeführt)
Bestellnummer: 6SN1197–0AA80–1AP1
DB1 Betriebsmeldungen/Alarmreaktionen
DD1 Diagnosefunktionen
DD2 Drehzahlregelkreis
DE1 Erweiterte Antriebsfunktionen
DF1 Freigaben
DG1 Geberparametrierung
DL1 MD des Linearmotors
DM1 Motor–/Leistungsteilparameter und Reglerdaten berechnen
DS1 Stromregelkreis
DÜ1 Überwachungen/Begrenzungen

/FBAN/ SINUMERIK 840D/SIMODRIVE 611 DIGITAL (Ausgabe 02.00)
Funktionsbeschreibung **ANA–MODUL**
Bestellnummer: 6SN1197–0AB80–0AP0

/FBD/ SINUMERIK 840D (Ausgabe 07.99)
Funktionsbeschreibung **Digitalisieren**
Bestellnummer: 6FC5297–4AC50–0AP0
DI1 Inbetriebnahme
DI2 Scan mit taktilem Sensor (scancad scan)
DI3 Scan mit Laser (scancad laser)
DI4 Fräsprogrammerstellung (scancad mill)

/FBDM/ SINUMERIK 840D/840Di/810D (Ausgabe 09.03)
Funktionsbeschreibung NC Programmmanagement
DNC Maschinen
Bestellnummer: 6FC5297–1AE81–0AP0

/FBDN/ SINUMERIK 840D/840Di/810D (Ausgabe 03.03)
Motion Control Information System (MCIS)
Funktionsbeschreibung **NC–Programmmagagement DNC**
Bestellnummer: 6FC5297–1AE80–0AP0
DN1 DNC Plant / DNC Cell
DN2 DNC IFC SINUMERIK, NC–Datenübertragung über Netzwerk

/FBFA/ SINUMERIK 840D/840Di/810D (Ausgabe 11.02)
Funktionsbeschreibung **ISO–Dialekte für SINUMERIK**
Bestellnummer: 6FC5297–6AE10–0AP3

/FBFE/ SINUMERIK 840D/810D (Ausgabe 03.04)
Funktionsbeschreibung **Ferndiagnose**
Bestellnummer: 6FC5297–0AF00–0AP3
FE1 Ferndiagnose (ReachOut)
FE3 RCS Host / RCS Viewer (pcAnywhere)

/FBH/
SINUMERIK 840D/840Di/810D
HMI–Programmierpaket (Ausgabe 11.02)
Bestellnummer: (ist Bestandteil der SW–Lieferung)

Teil 1 Benutzeranleitung
Teil 2 Funktionsbeschreibung

/FBH1/
SINUMERIK 840D/840Di/810D
HMI–Programmierpaket (Ausgabe 03.03)
ProTool/Pro Option SINUMERIK
Bestellnummer: (ist Bestandteil der SW–Lieferung)

/FBHL/
SINUMERIK 840D/SIMODRIVE 611 digital (Ausgabe 10.03)
Funktionsbeschreibung **HLA–Modul**
Bestellnummer: 6SN1197–0AB60–0AP3

/FBIC/
SINUMERIK 840D/840Di/810D (Ausgabe 06.03)
Motion Control Information System (MCIS)
Funktionsbeschreibung **TDI Ident Connection**
Bestellnummer: 6FC5297–1AE60–0AP0

/FBMA/
SINUMERIK 840D/810D (Ausgabe 08.02)
Funktionsbeschreibung **ManualTurn**
Bestellnummer: 6FC5297–6AD50–0AP0

/FBO/
SINUMERIK 840D/810D (Ausgabe 09.01)
Funktionsbeschreibung Projektierung **Bedienoberfläche OP 030**
(im folgenden sind die enthaltenen Kapitel aufgeführt)
Bestellnummer: 6FC5297–6AC40–0AP0
BA Bedienanleitung
EU Entwicklungsumgebung (Projektierpaket)
PSE Einführung in die Projektierung der Bedienoberfläche
(IK Installationspaket: Softwareupdate und Konfiguration)

/FBP/
SINUMERIK 840D (Ausgabe 03.96)
Funktionsbeschreibung **C–PLC–Programmierung**
Bestellnummer: 6FC5297–3AB60–0AP0

/FBR/
SINUMERIK 840D/840Di/810D (Ausgabe 01.04)
Funktionsbeschreibung **Rechnerkopplung RPC SINUMERIK**
Bestellnummer: 6FC5297–6AD61–0AP0
NFL Nahtstelle zum Fertigungsleitrechner
NPL Nahtstelle zu PLC/NCK

/FBSI/
SINUMERIK 840D/SIMODRIVE (Ausgabe 11.03)
Funktionsbeschreibung SINUMERIK **Safety Integrated**
Bestellnummer: 6FC5297–6AB80–0AP2

/FBSP/
SINUMERIK 840D/840Di/810D (Ausgabe 11.03)
Funktionsbeschreibung **ShopMill**
Bestellnummer: 6FC5297–6AD80–0AP2

/FBST/ SIMATIC (Ausgabe 01.01)
Funktionsbeschreibung **FM STEPDRIVE/SIMOSTEP**
Bestellnummer: 6SN1197–0AA70–0YP4

/FBSY/ SINUMERIK 840D/810D (Ausgabe 03.04)
Funktionsbeschreibung **Synchronaktionen**
Bestellnummer: 6FC5297–7AD40–0AP2

/FBT/ SINUMERIK 840D/810D (Ausgabe 03.04)
Funktionsbeschreibung **ShopTurn**
Bestellnummer: 6FC5297–6AD70–0AP2

/FBTC/ SINUMERIK 840D/810D (Ausgabe 01.02)
IT–Solutions
Funktionsbeschreibung **Tool Data Communication SinTDC**
Bestellnummer: 6FC5297–5AF30–0AP0

/FBTD/ SINUMERIK 840D/810D (Ausgabe 02.01)
IT–Solutions
Funktionsbeschreibung **Werkzeugbedarfsermittlung (SinTDI)**
mit Online–Hilfe
Bestellnummer: 6FC5297–6AE00–0AP0

/FBTP/ SINUMERIK 840D/840Di/810D (Ausgabe 01.03)
Motion Control Information System (MCIS)
Funktionsbeschreibung **Vorbeugende Instandhaltung TPM**
Bestellnummer: Dokument ist Bestandteil der Software

/FBU/ SIMODRIVE 611 universal/universal E (Ausgabe 07.03)
Funktionsbeschreibung **Regelungskomponente für Drehzahlregelung und Positionieren**
Bestellnummer: 6SN1197–0AB20–0AP8

/FBU2/ SIMODRIVE 611 **universal** (Ausgabe 04.02)
Montageanleitung
(liegt jedem SIMODRIVE 611 universal bei)

/FBW/ SINUMERIK 840D/810D (Ausgabe 11.02)
Funktionsbeschreibung **Werkzeugverwaltung**
Bestellnummer: 6FC5297–6AC60–0AP1

/HBA/ SINUMERIK 840D/840Di/810D (Ausgabe 03.02)
Handbuch **@Event**
Bestellnummer: 6AU1900–0CL20–0AA0

/HBI/ SINUMERIK 840Di (Ausgabe 09.03)
Handbuch **SINUMERIK 840Di**
Bestellnummer: 6FC5297–6AE60–0AP2

/INC/ SINUMERIK 840D840Di//810D (Ausgabe 06.03)
Systembeschreibung **Inbetriebnahme–Tool SINUMERIK SinuCOM NC**
Bestellnummer: (Bestandteil der Online–Hilfe des IBN–Tools)

/PJE/ SINUMERIK 840D/810D (Ausgabe 08.01)
Funktionsbeschreibung **Projektierpaket HMI Embedded**
Softwareupdate, Konfiguration, Installation
Bestellnummer: 6FC5297–6EA10–0AP0

/PS/ SINUMERIK 840D/810D (Ausgabe 09.03)
Projektierungsanleitung Projektiersyntax
Diese Schrift ist Bestandteil der SW–Lieferung und als PDF verfügbar

/POS1/ SIMODRIVE **POSMO A** (Ausgabe 08.03)
Benutzerhandbuch **Dezentraler Positioniermotor am PROFIBUS DP**
Bestellnummer: 6SN2197–0AA00–0AP6

/POS2/ SIMODRIVE **POSMO A** (Ausgabe 05.03)
Montageanleitung (liegt jedem POSMO A bei)

/POS3/ SIMODRIVE **POSMO SI/CD/CA** (Ausgabe 07.03)
Benutzerhandbuch **Dezentrale Servo Antriebstechnik**
Bestellnummer: 6SN2197–0AA20–0AP5

/POS4/ SIMODRIVE **POSMO SI** (Ausgabe 04.02)
Montageanleitung (liegt jedem POSMO SI bei)

/POS5/ SIMODRIVE **POSMO CD/CA** (Ausgabe 04.02)
Montageanleitung (liegt jedem POSMO CD/CA bei)

/S7H/ SIMATIC S7–300 (Ausgabe 2002)
Installationshandbuch **Technologische Funktionen**
Bestellnummer: 6ES7398–8AA03–8AA0
– Referenzhandbuch: CPU–Daten (HW–Beschreibung)
– Referenzhandbuch: Baugruppendaten

/S7HT/ SIMATIC S7–300 (Ausgabe 03.97)
Handbuch **STEP 7, Grundwissen, V. 3.1**
Bestellnummer: 6ES7810–4CA02–8AA0

/S7HR/ SIMATIC S7–300 (Ausgabe 03.97)
Handbuch **STEP 7, Referenzhandbücher, V. 3.1**
Bestellnummer: 6ES7810–4CA02–8AR0

/S7S/ SIMATIC S7–300 (Ausgabe 04.02)
Positionierbaugruppe **FM 353 für Schrittantrieb**
Bestellung zusammen mit dem Projektierpaket

/S7L/ SIMATIC S7–300 (Ausgabe 04.02)
Positionierbaugruppe **FM 354 für Servoantrieb**
Bestellung zusammen mit dem Projektierpaket

/S7M/ SIMATIC S7–300 (Ausgabe 01.03)
Mehrfachbaugruppe FM 357.2 für Servo– bzw. Schrittantrieb
Bestellung zusammen mit dem Projektierpaket

/SP/ SIMODRIVE 611–A/611–D
SimoPro 3.1
Programm zur Projektierung von Werkzeugmaschinen–Antrieben
Bestellnummer: 6SC6111–6PC00–0AAo
Bestellort: WK Fürth

d) Inbetriebnahme

/BS/ SIMODRIVE 611 analog (Ausgabe 10.00)
Beschreibung **Inbetriebnahmesoftware für Hauptspindel– und Asynchronmotormodule Version 3.20**
Bestellnummer: 6SN1197–0AA30–0AP1

/IAA/ SIMODRIVE 611A (Ausgabe 10.00)
Inbetriebnahmeanleitung
Bestellnummer: 6SN1197–0AA60–0AP6

/IAC/ SINUMERIK 810D (Ausgabe 11.02)
Inbetriebnahmeanleitung
(einschl. Beschreibung der Inbetriebnahme–Software SIMODRIVE 611D)
Bestellnummer: 6FC5297–6AD20–0AP1

/IAD/ SINUMERIK 840D/SIMODRIVE 611D (Ausgabe 03.04)
I**nbetriebnahmeanleitung**
(einschl. Beschreibung der Inbetriebnahme–Software SIMODRIVE 611D)
Bestellnummer: 6FC5297–7AB10–0AP0

/IAM/ SINUMERIK 840D/840Di/810D (Ausgabe 03.04)
Inbetriebnahmeanleitung **HMI**
Bestellnummer: 6FC5297–6AE20–0AP3
AE1 Aktualisierungen/Ergänzungen
BE1 Bedienoberfläche ergänzen
HE1 Online–Hilfe
IM2 Inbetriebnahme HMI Embedded
IM4 Inbetriebnahme HMI Advanced
TX1 Fremdsprachentexte erstellen mit Windows 95 / NT
TX2 Fremdsprachentexte erstellen mit Windows 2000 / XP

Platz für Notizen

Index

I

K

L

M

N

P

R

S

T

U

V

W

Z

Platz für Notizen

Befehle

Zeichen

Zahlen

A

B

D

G

H

K

L

M

An
SIEMENS AG
A&D MC BMS
Postfach 3180

D–91050 Erlangen

(Tel. 0180 / 5050 – 222 [Hotline]
Fax 09131 / 98 – 2176 [Dokumentation]
E–Mail: motioncontrol.docu@erlf.siemens.de)

Vorschläge

Korrekturen

für Druckschrift:

SINUMERIK 840D/840Di/810D
ISO–Dialekte für SINUMERIK
Funktionsbeschreibung

Hersteller–Dokumentation

Absender

Name

Anschrift Ihrer Firma/Dienststelle

Straße

PLZ: Ort:

Telefon: /

Telefax: /

Handbuch

Bestell–Nr.: 6FC5 297–6AE10–0AP4
Ausgabe: 09.04

Sollten Sie beim Lesen dieser Unterlage auf Druckfehler gestoßen sein, bitten wir Sie, uns diese mit diesem Vordruck mitzuteilen.
Ebenso dankbar sind wir für Anregungen und Verbesserungsvorschläge.

Vorschläge und/oder Korrekturen

Dokumentationsübersicht SINUMERIK 840D/840Di/810D (04.2004)

Allgemeine Dokumentation

SINUMERIK 840D/810D — Werbeschrift

SINUMERIK 840D/840Di/810D/ — Katalog Bestellu. NC 60 ***)**

Safety Integrated Applikationshandbuch — **Safety Interated** Applikations–handbuch

Anwender–Dokumentation

SINUMERIK 840D/810D/FM–NC — **AutoTurn**
- Kurzanleitung
- Programmieren /Einrichten

SINUMERIK 840D/840Di 810D — Bedienungsanl. **– HT 6**

SINUMERIK 840D/840Di/810D — **Diagnose–anleitung *)**

SINUMERIK 840D/840Di/810D — **Bedienungsanl. *)**
- Kurzanleitung
- HMI Embedded
- HMI Advanced

Anwender–Dokumentation

SINUMERIK 840D/840Di/810D — **Programmieranl.**
- Kurzanleitung
- Grundlagen ***)**
- Arbeitsvorbereit. ***)**
- Zyklen
- Meßzyklen
- ISO Turning/Milling

SINUMERIK 840D/810D — Bedienungsanl.
- **ManualTurn**
- Kurzanl. ManualTurn
- **ShopMill**
- Kurzanl. ShopMill
- **ShopTurn**
- Kurzanl. ShopTurn

SINUMERIK 840Di — **Systemüberblick**

SINUMERIK 840D/840Di/810D — Listen **Systemvariablen**

Hersteller–/Service–Dokumentation

SINUMERIK — **Projektierung** (HW) ***)**
- 810D
- 840D

SINUMERIK 840D/840Di 810D — **Bedien–komponenten** (HW) ***)**

SINUMERIK 840D/840Di/810D — Funktionsbeschr.
- **ManualTurn**
- **ShopMill**
- **ShopTurn**

Hersteller–/Service–Dokumentation

SINUMERIK SIMODRIVE 611D 840D/810D — Funktionsbeschr. **Synchronaktionen**

SINUMERIK 840D/840Di/810D — Funktionsbeschr. **Antriebsfunktion *)**

SINUMERIK 840D/840Di/810D — **Funktionsbeschr.**
- Grundmaschine ***)**
- Erweiterungsfunkt.
- Sonderfunktionen

SINUMERIK 840D/810D — Funktionsbeschr. **Werkzeug–verwaltung**

SINUMERIK 840D/810D — **Projektier–Pkt.** HMI Embedded

SINUMERIK 840D/810D — Funktionsbeschr. Projekt. Bedien–oberfläche **OP 030**

SINUMERIK 840D/810D — **MCIS**
- Rechnerkopplung
- Werkzeugbedarf
- NC–Datenverwaltung
- NC–Datenübertragung
- Tool Data Communication

Hersteller–/Service–Dokumentation

SINUMERIK SIMODRIVE — Funktionsbeschr. SINUMERIK **Safety Integrated**

SINUMERIK 840D — Funktionsbeschr. **Digitalisieren**

SINUMERIK SIMODRIVE — **Inbetriebnameanl. *)**
- 810D
- 840D/611D
- HMI

SINUMERIK SIMODRIVE 840D/840Di 810D 611D — **Listen *)**

SINUMERIK SIMODRIVE 840D 611D — Funktionsbeschr. **Linearmotor**

SINUMERIK SIMODRIVE 840D 611D — Funktionsbeschr.
- **Hydraulikmodul**
- **Analogmodul**

MOTION CONTROL SYSTEMS — **EMV**–Richtlinien

Elektronische Dokumentation

SINUMERIK SIMODRIVE 840D/840Di/810D 611, Motoren — **DOC ON CD *)** Das SINUMERIK–System

Hersteller–/Service–Dokumentation

SINUMERIK 840D/840Di/810D — Funktionsbeschr. **ISO–Dialekte** für SINUMERIK

SINUMERIK 840Di — **Handbuch** (HW + Inbetriebn.)

SINUMERIK 840D/810D — Funktionsbeschr. **Ferndiagnose**

SINUMERIK 840D/840Di/810D — **Handbuch @ Event**

***)** Empfohlener Minimalumfang der Dokumentation

Siemens AG
Automation and Drives
Motion Control Systems
Postfach 3180, D – 91050 Erlangen
Bundesrepublik Deutschland

www.siemens.de/motioncontrol

Bestell-Nr.: 6FC5297-6AE10-0AP4

Gedruckt in der Bundesrepublik Deutschland